UNIVERSITÉ DU QUÉBEC À MONTRÉAL

VÉRIFICATION DE MODÈLES FLOUE

MÉMOIRE
PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INFORMATIQUE

PAR
IVAN CONSTANTINEAU

JUILLET 2006

REMERCIEMENTS

Je tiens à remercier d'abord monsieur Guy Tremblay, Professeur au Département d'informatique, pour sa patience à mon égard, les nombreuses discussions que nous avons eues avant d'élire le sujet du mémoire et son aide précieuse dans le blitz final. C'est à sa suggestion que je me suis lancé dans l'aventure «floue». Je l'en remercie profondément. Ce sujet pertinent est encore plein de ressources à exploiter.

Je remercie le Syndicat des Chargées et Chargés de Cours de l'UQAM et l'Université du Québec à Montréal pour m'avoir octroyé la bourse qui m'a permis de mener à bien ce projet de maîtrise.

Je remercie aussi ma conjointe, Christine, qui a su braver les désagrements inhérents à l'accomplissement de ce mémoire.

TABLE DES MATIÈRES

RÉSUMÉ

Dans ce mémoire, on généralise la notion de vérification automatique de modèles au contexte flou. On définit des structures de Kripke floues et on leur associe des logiques temporelles floues, dénotées $\aleph CTL^*$ et $\aleph CTL$. On vérifie que les opérateurs de la logique $\aleph CTL$ sont monotones et qu'il y a moyen de faire de la vérification de modèles dans ce contexte. On en fait alors la démonstration.

CHAPITRE I

INTRODUCTION

Ce mémoire traite de *vérification de modèles*, terme traduit de l'anglais *Model Checking* et dont on pourra trouver les rudiments dans (15), (3) ou (14). Nous en revoyons les aspects essentiels sous l'angle de la *logique floue*. Suivant la terminologie employée dans (11) nous abordons le sujet dans le cadre restreint de la vérification de modèles *globale* avec une approche dite *sémantique* ou *itérative*. Sous ce dernier aspect, notre travail rejoint l'article de Chechik *et al.* (2) qui consiste en une généralisation de la vérification de modèles basée sur des logiques multivalentes finies.

Pour faire un peu de (petite) histoire disons que nous avons commencé le présent mémoire en tentant d'adapter ce dernier travail au cas flou, infini. Cette entreprise semblait raisonnable dans le cadre d'une maîtrise en informatique théorique. Une fois le travail complété, ce qui ne s'est pas fait sans peine, nous avons pressenti et réalisé le fait qu'il n'était peut-être pas nécessaire de procéder comme il est fait dans (2).

Dans ce dernier article on généralise les opérateurs CTL au contexte multivalent en y étendant d'abord la définition de la modalité **EX**. Pour ce faire, on introduit une opération sur des ensembles multivalués que l'on nomme «*Backward image*» qui permet de définir la sémantique de l'opérateur CTL multivalué **EX**. À l'instar de ce qui est fait dans (3), on exécute cette opération de manière groupée plutôt que de l'interpréter comme la généralisation d'une possible composition d'opérateurs distincts, notamment **E** et **X**, comme cela est présenté, par exemple, dans (3). Le résultat est une logique appelée χCTL.

Mais, même en contexte multivarié, il nous semblait plus naturel de calculer la sémantique de ces deux opérateurs de manière totalement indépendante l'une de l'autre. Cette perspective ne coïncidait pas vraiment avec le caractère «monolitique» des opérateurs «*Backward*».

Nous avons choisi plutôt de généraliser la logique CTL^* (où les opérateurs **E** et **X** cohabitent de manière strictement indépendante l'un de l'autre), et tous les opérateurs qui la composent, à la logique floue standard.

Là encore, le tout ne s'est fait sans embûches. En effet, bien que certains opérateurs se généralisent aisément en logique floue, d'autres, comme l'opérateur *Until*, noté **U** dans le mémoire, demandent un soin particulier car on peut les définir de plusieurs manières possibles.

Par exemple, dans le cas précis de l'opérateur **U**, défini ainsi :

- *pour f et g des propriétés d'un système donné, l'opération (f U g) signifie que f est vraie jusqu'à ce que g le soit, en précisant que g le sera bel et bien un jour*

on pourrait être porté à croire que l'aspect principal de cet opérateur réside dans le fait que la propriété g est vraie pour une *première* fois, car alors, à l'instant où g est vraie, il faut que f ait toujours été vraie avant cet instant pour que $(f\mathbf{U}g)$ soit vraie.

Or ce n'est pas le cas. En fait, l'idée fondamentale qui permet de généraliser efficacement l'opérateur **U** ne tient pas tant dans le fait qu'il y a une première fois où la propriété g est vraie comme dans le fait que «pour avoir $(f\mathbf{U}g)$ vraie»:

il y a une première fois où la propriété g est vraie et elle peut aussi l'être subséquemment.

Notons que cette propriété banale a peu, voire pas du tout d'incidence sur la version ordinaire de l'opérateur **U**. La logique binaire *masque* l'intérêt fondamental qu'elle a en logique floue. En effet, cette dernière apporte ici une nuance importante en donnant à l'interprétation finale du résultat une valeur intermédiaire entre vrai et le faux.

Ainsi, dans le cas qui nous occupe, l'évaluation de l'opération floue $(f\mathbf{U}g)$ sur un chemin π dépend de celle des valeurs de vérités des formules f et g sur π qui peuvent non seulement être 0 ou 1, mais aussi prendre toute valeur entre 0 et 1. Cela a pour conséquence que l'on doit, en général, tenir compte de l'évaluation de la valeur de vérité de la formule g sur un nombre plus grand d'états que dans le cas ordinaire. En fait, l'évaluation de la version floue de $f\mathbf{U}g$ repose, nous le verrons plus loin, sur des opérations consistant à trouver le maximum et/ou le minimum de certains ensembles de valeurs de vérités.

Et nous somme tombés dans le piège! Nous avons d'abord défini inadéquatement l'opération

$f\mathrm{U}g$ en tenant compte du fait que l'opération g était «vraie» une *première* fois. Cela nous a mené à définir une généralisation complexe et tordue (dans le sens mathématique — «*skewed*» si on préfère). Le travail avançait lentement et chaque identité usuelle de CTL^* devait être revue complètement pour faire en sorte que sa généralisation ait un sens.

Puis la lumière fut. Révisant une n+unième fois notre ébauche, nous y avons (la +unième fois) trouvé une erreur de fond salutaire, nous forçant à revoir complètement la définition de base de l'opérateur *Until*. Salutaire, en effet, car nous avons découvert que la meilleure généralisation de cet opérateur est aussi la plus simple qui soit dans le contexte : il n'est pas *nécessaire* de comptabiliser le *premier* instant où la seconde propriété est vraie.

D'un point de vue logique (temporelle), tout devient clair si on fait ce constat. Il est possible alors d'évaluer la sémantique de l'opération *floue* $f\mathrm{U}g$ uniquement avec les généralisations ordinaires des connecteurs et quantificateurs logiques calculés à l'aide des fonctions max et min. Cette généralisation coïncide parfaitement avec la définition de l'opération ordinaire.

Bien sûr, il a fallu tout recommencer une autre fois! Mais cela a été la partie facile. La généralisation finale que nous proposons et étudions dans ce mémoire est naturelle. Les identités que l'on trouve habituellement dans CTL^* se relèvent aisément. On pourrait même penser qu'il s'agit d'une simple traduction «catégorique» voire «fonctorielle» de la vérification de modèles ordinaire tant, une fois située dans le «bon» contexte, elles sont identiques.

Il faut cependant prendre garde de ne pas sous-estimer cette généralisation. Par exemple, dans ce nouveau contexte, le travail de Chechik *et al.* dans (2) se résume, théoriquement parlant du moins, à l'étude d'un opérateur particulier que l'on peut introduire indépendamment des autres opérateurs flous. Nous le verrons dans le chapitre 3.

Nous avons découpé le corps du mémoire en trois chapitres. Dans le chapitre 2, nous introduisons en premier lieu les outils de base permettant de saisir le contexte dans lequel vit la vérification de modèles, à savoir la théorie de treillis complets et les systèmes de transition. Puis nous décrivons la méthode itérative classique permettant de résoudre le problème de la vérification de modèles ordinaire.

Nous expliquons dans le chapitre 3 d'abord les rudiments de la logique floue nécessaires à la conception de la méthode itérative floue solutionnant la vérification de modèles floue. Puis nous en démontrons la viabilité.

Dans le dernier chapitre, nous survolons le fait que la vérification de modèles floue est possible sans toutefois vraiment creuser la question. Nous terminons le mémoire en donnant des avenues de recherches futures dans la conclusion.

Il importe d'ailleurs de préciser ici les limites de cet ouvrage. Comme il a été souligné au départ, nous ne visons qu'à montrer la possibilité (théorique) de généraliser la vérification de modèles ordinaire au cas où on se sert de logiques infinies (floues) pour en décrire les propriétés. Aucune mise en oeuvre de ces recherches n'a été tentée jusqu'à maintenant, en ce qui nous concerne. De même, la relation avec le monde flou, tourné systématiquement vers les applications *concrètes*, est relativement ténue. La logique floue apparaît en fait parce qu'on y traite de logiques de valence infinie.

Cependant, à notre connaissance, ce terrain de recherche est encore vierge à toutes fins pratiques. Sur «*Google*» les mots clés «*"fuzzy logic", "model checking"*» aboutissaient, au moment où nous avons exploré la chose, systématiquement aux travaux de Chechik *et al.* où on précise de manière explicite ne pas aborder les logiques infinies. Les seules recherches bibliographiques positives que nous avons obtenues nous ont conduit à des vérifications de modèles probabilistes qui ont peu en commun avec ce que nous avons fait.

CHAPITRE II

LES STRUCTURES MATHÉMATIQUES

La vérification de modèles globale consiste à déterminer pour un modèle fini M d'un système donné et pour une fomule ϕ donnée, l'ensemble des états de M qui satisfont ϕ. Pour y parvenir, nous procédons itérativement et obtenons cet ensemble comme point fixe (plus petit ou plus grand) d'un opérateur particulier. Le contexte mathématique général approprié pour introduire ces opérateurs est celui des *treillis complets*. On peut en effet y définir la notion d'opérateur monotone avec laquelle l'existence d'un point fixe est toujours assurée. Nous étudions sommairement ces treillis dans la section 2.1 et terminons l'étude plus spécifique des points fixes dans la section 2.2.

Dans ce mémoire, le terme *système* est entendu dans le sens informel de *système d'information*. Cependant, le concept mathématique précis qui modélise cette dernière notion et que nous reprenons ici, est celui de *système de transition*, plus précisément celle de *structure de Kripke*. Il fait l'objet de la section 2.3.

Les propriétés des systèmes sont, quant à elles, exprimées à l'aide de logiques modales spéciales, dites des *logiques temporelles*. Après les avoir brièvement décrites en général nous donnons, dans la section 2.4, deux versions de la logique CTL^*. La seconde de ces versions est celle qui nous permettra de définir la logique floue analogue, $\aleph CTL^*$, au chapitre 3. Nous terminons ce chapitre avec l'étude des propriétés itératives de CTL qui permettent de démontrer que l'on peut faire de la vérification de modèles avec CTL, le but final de ce travail étant de montrer qu'il est possible de démontrer la même chose avec la logique $\aleph CTL^*$.

2.1 Les treillis

Définition 1 Soit P un ensemble. Un *ordre partiel* sur P est un sous-ensemble du produit cartésien de P avec lui-même, appelé aussi une relation binaire sur P, dénotée sous forme d'opérateur infixe $\leq$, et satisfaisant les conditions suivantes. Pour tout x, y, z dans P on a:

i. $x \leq x$, (réflexivité)

ii. $x \leq y$ et $y \leq x$ entraînent $x = y$, (antisymétrie)

iii. $x \leq y$ et $y \leq z$ entraînent $x \leq z$, (transitivité).

Remarque 1 Lorsqu'il n'y a pas de confusion possible, on désigne un ordre partiel sur un ensemble P comme étant P lui-même, au lieu d'utiliser la notation structurelle $< P, \leq >$. Un ordre partiel est aussi appelé un *ordre* tout court. Dans un ordre partiel P, il se peut que deux éléments $x, y \in P$ ne soient pas comparables, c'est-à-dire que ni $x \leq y$, ni $y \leq x$ ne soit vrai. Lorsqu'on exige qu'ils le soient, l'ordre est dit *total.*

Exemples :

1. Soit E un ensemble. On désigne par $\wp(E)$ l'ensemble des parties de E. Il est bien connu que la relation d'inclusion $\subseteq$ est un ordre partiel sur $\wp(E)$.

2. Le segment de nombre réels $[0, 1]$ muni de l'ordre usuel sur les nombres réels "$\leq$" est un ordre total.

Définition 2 Soit P un ordre partiel et soit $Q \subseteq P$. Soit $a \in Q$. On dit que:

i. a est un *élément maximal* de Q si, pour tout $x \in Q$ on a $a \leq x \Rightarrow a = x$ et a est un *élément minimal* de Q si, pour tout $x \in Q$ on a $x \leq a \Rightarrow a = x$.

ii. a est le *plus grand élément* de Q si, pour tout $x \in Q$ on a $x \leq a$ et a est le *plus petit élément* de Q si, pour tout $x \in Q$ on a $a \leq x$.

Définition 3 Soit P un ensemble ordonné. Le plus grand élément de P, s'il existe, est dénoté $\top$, le *plafond* de P et le plus petit élément de P, s'il existe, est dénoté $\bot$, le *plancher* de P.

Exemples :

1. Le plancher et le plafond de $\wp(E)$ sont respectivement l'ensemble vide, $\emptyset$, et l'ensemble E lui-même.

2. Le plancher et le plafond de $[0,1]$ sont respectivement 0 et 1.

Définition 4 Soit P un ensemble ordonné et un sous-ensemble $S \subseteq P$. Un élément $x \in P$ est une *borne supérieure* de S si, pour tout $s \in S$, on a $s \leq x$. Un élément $y \in P$ est une *borne inférieure* de S si, pour tout $s \in S$, on a $y \leq s$. L'ensemble des bornes supérieures de S est dénoté par $\sigma(S)$ et celui des bornes inférieures de S par $\iota(S)$. Si l'ensemble $\sigma(S)$ admet un plus petit élément $x \in P$ alors x est appelé la *plus petite borne supérieure* et est dénoté $\sup(S)$. Si l'ensemble $\iota(S)$ admet un plus grand élément $y \in P$ alors y est appelé la *plus grande borne inférieure* et est dénoté $\inf(S)$. On dénote par $x \wedge y = \inf(\{x,y\})$, lorsque cet inf existe et $x \vee y = \sup(\{x,y\})$, lorsque ce sup existe.

Définition 5 Un *treillis* sur un ensemble A est un ordre $(A, \leq)$ tel que pour toute paire d'éléments x, y dans A, $x \wedge y$ et $x \vee y$ existent. Un treillis est *complet* lorsque pour tout sous-ensemble B d'éléments de A, on a que $\sup(B)$ et $\inf(B)$ existent dans A.

Proposition 1 *Les ordres $\wp(E)$ et $[0,1]$ sont des treillis complets.*

Si $< A, \leq >$ est un ordre sur A muni d'une structure de treillis, alors les lois suivantes sont valides dans A:

i. Idempotence

Pourtout $x \in A$, $x \wedge x = x$ et $x \vee x = x$.

ii. Commutativité

Pourtout $x, y \in A$, $x \wedge y = y \wedge x$ et $x \vee y = y \vee x$.

iii. Associativité

Pourtout $x, y, z \in A$, $(x \wedge y) \wedge z = x \wedge (y \wedge z)$ et $(x \vee y) \vee z = x \vee (y \vee z)$.

iv. Absorption

Pourtout $x, y \in A$, $x \vee (x \wedge y) = x$ et $x \wedge (x \vee y) = x$.

v. Neutre

Pourtout $x \in A,\ x \vee \bot = \bot \vee x = x$ et $x \wedge \top = \top \wedge x = x$.

Définition 6 Un treillis $< A, \leq >$ est *distributif* si et seulement si, pour tout x, y, z dans A on a

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z) \text{ et } x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z).$$

Définition 7 Un treillis $< A, \leq >$ est *booléen* si et seulement si,

- il existe un plancher et un plafond dans A,
- le treillis $< A, \leq >$ est distributif,
- pour tout élément $a \in A$ il existe un élément $\sim a$ tel que $a \wedge \sim a = \bot$ et $a \vee \sim a = \top$. L'élément $\sim a$ est appelé le *complément* de a.

Les treillis booléens sont mieux connus sous le nom d'*algèbres de Boole*, le terme "algèbre" étant entendu dans le sens des algèbres universelles.

Proposition 2 *Dans toute algèbre de Boole, on a, pour tout* $a,\ b$ *dans* A,

i. $\sim \bot = \top$ et $\sim \top = \bot$

ii. $\sim (\sim a) = a$

iii. $a \vee \sim a = \top$ *et* $a \wedge \sim a = \bot$

iv. Les lois de De Morgan:

a. $\sim (a \wedge b) = \sim a \vee \sim b$

b. $\sim (a \vee b) = \sim a \wedge \sim b$

Exemple :

On dénote par **B** l'algèbre de Boole contenant les valeurs logiques **true** et **false**, muni des opérations "**et**" logique, pour le $\wedge$ et le "**ou**" logique, pour le $\vee$. On utilisera aussi l'algèbre de Boole $\mathbf{B}'$ contenant les deux entiers 0 et 1, muni des opérations addition modulo 2 pour le $\vee$ et de multiplication usuelle pour le $\wedge$. Ces deux algèbres sont isomorphes.

Définition 8 Soient les opérations binaires

$$\text{Max} : [0,1] \times [0,1] \to [0,1] \text{ et } \text{Min} : [0,1] \times [0,1] \to [0,1]$$

telles que pour tout x, y dans $[0,1]$, $\text{Max}(x,y)$ donne le plus grand des deux nombres x ou y ordonnés selon l'ordre usuel et tel que tout x, y dans $[0,1]$, $\text{Min}(x,y)$ donne le plus petit des deux nombres x ou y ordonnés selon l'ordre usuel. On définit les opérateurs infixes correspondants

$$\text{'Max'} : [0,1] \times [0,1] \to [0,1] \text{ et } \text{'Min'} : [0,1] \times [0,1] \to [0,1]$$

où, pour tout x, y dans $[0,1]$, on a respectivement x 'Max' $y = \text{Max}(x,y)$ et x 'Min' $y = \text{Min}(x,y)$. Si $E = \{x_1,\ x_2, \ldots x_n\}$ est un ensemble fini et $f : E \to [0,\ 1]$ une fonction, alors on définit deux quantificateurs numériques $\max\limits_{x \in E} f(x)$ et $\min\limits_{x \in E} f(x)$ respectivement par les formules suivantes:

$$\max_{x \in E} f(x) = f(x_1) \text{ 'Max' } f(x_2) \text{ 'Max' } \ldots \text{'Max' } f(x_n),$$

$$\min_{x \in E} f(x) = f(x_1) \text{ 'Min' } f(x_2) \text{ 'Min' } \ldots \text{'Min' } f(x_n).$$

Par convention, lorsque E est vide, le quantificateur retourne le neutre de l'opération itérée. Ainsi, puisque f prend ses valeurs dans l'intervalles $[0,\ 1]$, on a

$$\max_{x \in \emptyset} f(x) = 0,$$

$$\min_{x \in \emptyset} f(x) = 1.$$

Proposition 3 *Pour tout entier $n \geq 0$, tout ensemble $E = \{x_1,\ x_2, \ldots x_n\}$ et toute fonction $f : E \to [0,\ 1]$, on a*

$$1 - \max_{x \in E} f(x) = \min_{x \in E}(1 - f(x))$$

et

$$1 - \min_{x \in E} f(x) = \max_{x \in E}(1 - f(x)).$$

Démonstration. Pour la première identité, notons que la fonction $\varphi(x) = 1-x$ est décroissante car si $x \leq y$ alors $-x \geq -y$ et $1-x \geq 1-y$. Supposons x_i tel que $f(x_i) = \max_{x \in E} f(x)$ alors pour tout $y \in E$ on a $f(x_i) \geq f(y)$ et ainsi

$$1 - \max_{x \in E}(f(x)) = 1 - f(x_i) = \min_{y \in E}(1 - f(y)).$$

La seconde indentité se prouve de la même manière.

2.2 La théorie des points fixes

Définition 9 Soit S un ensemble. Un *prédicat* f sur S est une fonction $f : S \to \mathbf{B}$.

Soit S un ensemble. On peut considérer tout sous-ensemble $S' \in \wp(S)$ comme un *prédicat* sur S avec $S' : S \to \mathbf{B}$, en posant:

$$\forall x \in S,\ S'(x) = \mathbf{vrai} \Leftrightarrow x \in S'.$$

On distingue deux prédicats particuliers sur S dénotés $\mathbf{Vrai}_S$ et $\mathbf{Faux}_S$, associés respectivement à l'ensemble S et à l'ensemble vide, définis comme suit:

$$\forall x \in S,\ \mathbf{Vrai}_S(x) = \mathbf{vrai}$$

et

$$\forall x \in S,\ \mathbf{Faux}_S(x) = \mathbf{faux}.$$

On dénote aussi ces derniers respectivement par **Vrai** et **Faux** lorsqu'aucune confusion n'est à craindre.

Définition 10 Une fonction $\tau : \wp(S) \to \wp(S)$ est appelée une *transformation de prédicats* sur S.

Définition 11 Soit S un ensemble et $\tau : \wp(S) \to \wp(S)$ une transformation de prédicats sur S.

i. On dit que τ est *monotone* si et seulement si pour tous sous-ensembles P, Q dans $\wp(S)$ on a

$$P \subseteq Q \Rightarrow \tau(P) \subseteq \tau(Q).$$

ii. On dit que τ est $\cup$-*continue* si et seulement si pour tous sous-ensembles $P_1, P_2, P_3 \ldots$ dans $\wp(S)$ on a

$$P_1 \subseteq P_2 \subseteq P_3 \ldots \Rightarrow \tau(\cup_i P_i) = \cup_i \tau(P_i).$$

iii. On dit que τ est $\cap$-*continue* si et seulement si pour tous sous-ensembles $P_1, P_2, P_3 \ldots$ dans $\wp(S)$ on a

$$P_1 \supseteq P_2 \supseteq P_3 \ldots \Rightarrow \tau(\cap_i P_i) = \cap_i \tau(P_i).$$

iv. On dénote les *itérées* de τ sous la composition usuelle par $\tau^i, i \geq 0$ où on définit, pour tout $Z \in \wp(S)$, $\tau^0(Z) = Z$ et pour tout $i \geq 0$, $\tau^{i+1}(Z) = \tau(\tau^i(Z))$.

v. Tout sous-ensemble $Z \in \wp(S)$ est appelé un *point fixe* de τ si et seulement si on a

$$\tau(Z) = Z.$$

vi. Le *plus petit* point fixe de τ, s'il existe, est le sous-ensemble $Z \in \wp(S)$ tel que

a. Z est un point fixe de τ,

b. si Y est un point fixe de τ, alors $Z \subseteq Y$.

On le dénote par $\mu Z.\tau(Z)$.

vii. Le *plus grand* point fixe de τ, s'il existe, est le sous-ensemble $Z \in \wp(S)$ tel que

a. Z est un point fixe de τ,

b. et si Y est un point fixe de τ, alors $Z \supseteq Y$.

On le dénote par $\nu Z.\tau(Z)$.

Théorème 1 (Knaster-Tarski) *Soit S un ensemble et τ une transformation de prédicats sur S. Si τ est monotone, alors on a:*

i. $\mu Z.\tau(Z) = \bigcap \{Z | \tau(Z) \subseteq Z\}$

ii. $\nu Z.\tau(Z) = \bigcup \{Z | \tau(Z) \supseteq Z\}$

Démonstration. Pour démontrer l'assertion *i.*, posons

$$\Phi = \{X | \tau(X) \subseteq X\} \quad \text{et} \quad Y = \bigcap_{X \in \Phi} X.$$

Nous devons montrer que Y est la plus petite solution de l'équation

$$\tau(X) = X.$$

Notons d'abord que Φ n'est pas vide car on a $\tau(S) \subseteq S$ ce qui entraîne que S est dans Φ. Remarquons aussi que si on a un sous-ensemble $X \in \wp(S)$ tel que $\tau(X) = X$, alors on a $\tau(X) \subseteq X$. Montrons que $Y = \tau(Y)$.

Pour tout X dans Φ on a l'inclusion $Y \subseteq X$ car l'intersection définissant Y se fait aussi sur X. Puisque τ est monotone, on a $\tau(Y) \subseteq \tau(X) \subseteq X$ ce qui fait que $\tau(Y) \subseteq X$. Cela étant vrai quelque soit X dans Φ, cela est aussi vrai de leur intersection, $\cap_{X \in \Phi} X$. On a donc $\tau(Y) \subseteq Y$.

Puisque τ est monotone, on a alors

$$\tau(\tau(Y)) \subseteq \tau(Y),$$

c'est-à-dire que $\tau(Y) \in \Phi$. Donc, puisque Y est sous-ensemble de chaque élément de Φ on a bien $Y \subseteq \tau(Y)$. L'assertion *i.* est démontrée.

Pour démontrer la seconde assertion, il suffit de «dualiser» la preuve précédente. C'est-à-dire que si, dans un énoncé, on échange respectivement S, $\subset$ et $\cap$ par $\emptyset$, $\supset$ et $\cup$ alors les axiomes des algèbres de Boole sont invariants. Un argument basé sur ces transformations sera donc toujours valide une fois ces dernières faites. Posons

$$\Phi = \{X | X \subseteq \tau(X)\} \quad \text{et} \quad Y = \bigcup_{X \in \Phi} X.$$

Nous devons montrer que Y est la plus grande solution de l'équation

$$\tau(X) = X.$$

Notons d'abord que Φ n'est pas vide car on a $\tau(\emptyset) \supseteq \emptyset$ ce qui entraîne que $\emptyset$ est dans Φ. Remarquons aussi que si on a un sous-ensemble $X \in \wp(S)$ tel que $\tau(X) = X$, alors on a $\tau(X) \supseteq X$. Montrons que $Y = \tau(Y)$.

Pour tout X dans Φ on a l'inclusion $Y \supseteq X$ car la réunion définissant Y se fait aussi sur X. Puis que τ est monotone, on a $\tau(Y) \supseteq \tau(X) \supseteq X$ ce qui fait que $\tau(Y) \supseteq X$. Cela étant vrai quelque soit X dans Φ, cela est aussi vrai de leur réunion, $\cup_{X \in \Phi} X$. On a donc $\tau(Y) \supseteq Y$.

Puisque τ est monotone, on a alors

$$\tau(\tau(Y)) \supseteq \tau(Y),$$

c'est-à-dire que $\tau(Y) \in \Phi$. Donc, puisque Y est sur-ensemble de chaque élément de Φ on a bien $Y \supseteq \tau(Y)$. L'assertion *ii.* est démontrée.

Lemme 1 *Soit S un ensemble. Alors toute transformation de prédicats τ qui est $\cup$-continue ou $\cap$-continue est monotone. De plus, si S est fini alors toute transformation monotone sur S est $\cup$-continue et $\cap$-continue.*

Démonstration. Soient X, X' des sous-ensembles de S tels que $X \subseteq X'$. Alors on a

$$X \cup X' = X', \text{ et } X \cap X' = X.$$

Si τ est $\cup$-continue alors on a $\tau(X') = \tau(X) \cup \tau(X')$ et si τ est $\cap$-continue alors on a $\tau(X) = \tau(X) \cap \tau(X')$. Dans les deux cas on a bien $\tau(X) \subseteq \tau(X')$ ce qui achève la démonstration de la première assertion.

Pour démontrer la seconde, supposons S de cardinalité finie. Soit $(X_i)_{i\geq 0}$ une suite de sous-ensembles croissante de S, c'est-à-dire

$$X_0 \subseteq X_1 \subseteq X_2 \subseteq \ldots$$

et telle que pour tout $i \in \mathbf{N}$ on a $|X_i| = n_i$. Alors on a

$$n_0 \leq n_1 \leq n_2 \ldots \leq |S|.$$

Cela signifie qu'il existe un entier $j \geq 0$ tel que pour tout entier $k \geq j$, on a $n_k = n_j$ ce qui entraîne à son tour que pour tout entier $k \geq j$ on a $X_k = X_j$. Puisque la suites des X_i est croissante on a bien

$$\bigcup_{k\geq 0} X_k = X_j \text{ et } \tau(\bigcup_{k\geq 0} X_k) = \tau(X_j).$$

D'autre part, τ est monotone. Donc, puisque la suite $(X_i)_{i\geq 0}$ est croissante, on a

$$\tau(X_0) \subseteq \tau(X_1) \subseteq \tau(X_2) \ldots \subseteq \tau(X_j) \ldots$$

où, pour tout $k < j$ on a $\tau(X_k) < \tau(X_j)$ et, pour tout $k \geq j$, on a $\tau(X_k) = \tau(X_j)$. Donc

$$\bigcup_{k\geq 0} \tau(X_k) = \tau(X_j) \text{ et } \tau(\bigcup_{k\geq 0} X_k) = \bigcup_{k\geq 0} \tau(X_k).$$

Théorème 2 (Théorème du point fixe de Kleene) *Si τ est $\cup$-continue (respectivement $\cap$-continue), alors on a*

$$\mu Z.\tau(Z) = \bigcup_{i\geq 0} \tau^i(\emptyset) \text{ (resp. } \nu Z.\tau(Z) = \bigcap_{i\geq 0} \tau^i(S)).$$

Démonstration. Soit une transformation de prédicats τ $\cup$-continue. Montrons, par récurrence, que τ est croissante. On a, par définition d'itérée, $\emptyset = \tau^0(\emptyset)$. Donc, évidemment, $\tau^0(\emptyset) \subseteq \tau^1(\emptyset)$. Par le lemme précédent, τ est monotone. Ainsi on a

$$\forall i \geq 0, \quad \tau^i(\emptyset) \subseteq \tau^{i+1}(\emptyset) \Rightarrow \tau(\tau^i(\emptyset)) = \tau^{i+1}(\emptyset) \subseteq \tau(\tau^{i+1}(\emptyset)) = \tau^{i+2}(\emptyset).$$

Puisque τ est $\cup$-continue on a

$$\tau(\bigcup_{i\geq 0} \tau^i(\emptyset)) = \bigcup_{i\geq 0} \tau^{i+1}(\emptyset) = \bigcup_{i\geq 0} \tau^i(\emptyset).$$

C'est-à-dire que $\bigcup_{i\geq 0} \tau^i(\emptyset)$ est un point fixe de τ. Pour voir que c'est le plus petit point fixe, supposons $X \in \wp(S)$ un point fixe de τ. Alors on a $\emptyset = \tau^0(\emptyset) \subseteq X$. De plus, pour tout entier $i \geq 0$ on a $\tau^i(\emptyset) \subseteq X \Rightarrow \tau^{i+1}(\emptyset) \subseteq \tau(X) = X$. Donc $\bigcup_{i\geq 0} \tau^i(\emptyset) \subseteq X$.

La première partie du théorème est démontrée. On démontre la seconde partie en dualisant la première.

Montrons, par récurrence, que τ est décroissante. On a, par définition d'itérée, $S = \tau^0(S)$. Donc, évidemment, $\tau^0(S) \supseteq \tau^1(S)$. Par le lemme précédent, τ est monotone. Ainsi on a

$$\forall i \geq 0, \quad \tau^i(S) \supseteq \tau^{i+1}(S) \Rightarrow \tau(\tau^i(S)) = \tau^{i+1}(S) \supseteq \tau(\tau^{i+1}(S)) = \tau^{i+2}(S).$$

Puisque τ est $\cap$-continue on a

$$\tau(\bigcap_{i\geq 0} \tau^i(S)) = \bigcap_{i\geq 0} \tau^{i+1}(S) = \bigcap_{i\geq 0} \tau^i(S).$$

C'est-à-dire que $\bigcap_{i\geq 0} \tau^i(S)$ est un point fixe de τ. Pour voir que c'est le plus grand point fixe, supposons $X \in \wp(S)$ un point fixe de τ. Alors on a $S = \tau^0(S) \supseteq X$. De plus, pour tout entier $i \geq 0$ on a $\tau^i(S) \supseteq X \Rightarrow \tau^{i+1}(S) \supseteq \tau(X) = X$. Donc $\bigcap_{i\geq 0} \tau^i(S) \supseteq X$. Le théorème est démontré.

Théorème 3 *Soit τ une transformation de prédicats monotone sur S, où S est un ensemble de cardinalité finie, $n = |S|$. Alors on a*

$$\mu Z.\tau(Z) = \tau^n(\emptyset) \quad \text{et} \quad \nu Z.\tau(Z) = \tau^n(S).$$

Démonstration. La transformation τ est monotone. Par le lemme 1, puisque S est fini, elle est $\cup$-continue et $\cap$-continue. Ainsi, par le théorème 2 on a

$$\mu Z.\tau(Z) = \bigcup_{i\geq 0} \tau^i(\emptyset) \text{ (resp. } \nu Z.\tau(Z) = \bigcap_{i\geq 0} \tau^i(S)).$$

Soit n_i le nombre d'éléments dans $\tau^i(\emptyset)$ (resp. $n_i = |S| - |\tau^i(S)|$). Puisque τ est monotone et S fini, la suite (n_k) est croissante (resp. décroissante) majorée par n (resp. minorée par 0). Soit j le plus petit entier tel que $n_j = n_{j+1}$. Alors $\tau^j(\emptyset) = \tau^{j+1}(\emptyset)$ (resp. $\tau^j(S) = \tau^{j+1}(S)$) car ces deux ensembles ont le même nombre d'éléments et que le premier est inclus dans le second. Cela entraîne

$$\tau^{j+2}(\emptyset) = \tau(\tau^{j+1}(\emptyset)) = \tau(\tau^j(\emptyset)) = \tau^{j+1}(\emptyset) \quad (\text{resp. } \tau^{j+2}(S) = \tau(\tau^{j+1}(S)) = \tau^{j+1}(S)).$$

Ainsi $t^n(\emptyset) = t^j(\emptyset)$ (resp. $t^n(S) = t^j(S)$). Puisque, pour tout entier $0 \leq i \leq j$, on a $t^i(\emptyset) \subseteq t^j(\emptyset)$ (resp. $t^i(S) \supseteq t^j(S)$) alors on a

$$\tau^n(\emptyset) = \tau^j(\emptyset) = \bigcup_{i\geq 0} \tau^i(\emptyset) \quad (\text{resp. } \tau^n(S) = \tau^j(S) = \bigcap_{i\geq 0} \tau^i(S)).$$

2.3 Les systèmes de transitions

Définition 12 Soit S un ensemble (fini ou infini). Un système de transition *simple* sur S est la donnée d'un quadruplet

$$\mathbf{A} = < S,\ T,\ \alpha,\ \beta >$$

où

- S est appelé l'ensemble des *états* de $\mathbf{A}$
- T est l'ensemble des *transitions* de $\mathbf{A}$ telles que:
  a. $\alpha : T \to S$ est une fonction qui assigne à chaque transition t son *origine* $\alpha(t)$,
  b. $\beta : T \to S$ est une fonction qui assigne à chaque transition t son *but* $\beta(t)$.

Définition 13 Soit S un ensemble (fini ou infini). Un système de transition *étiqueté* sur S est la donnée d'un sextuplet

$$\mathbf{A} = < S,\ T,\ A,\ \alpha,\ \beta,\ \gamma >$$

où

- $< S,\ T,\ \alpha,\ \beta >$ est un système de transition simple sur S,
- A est un ensemble fini appelé *l'alphabet* de $\mathbf{A}$,
- $\gamma : T \rightarrow A$ est la fonction d'*étiquetage* des transitions de $\mathbf{A}$.

Définition 14 Soit S un ensemble, E et F deux ensembles finis identifiés respectivement comme les *paramètres d'états* et les *paramètres de transitions*. Un système de transition *paramétré* par E et F sur S est la donnée d'un octotuplet

$$\mathbf{A} = < S,\ T,\ E,\ F,\ \alpha,\ \beta,\ \gamma,\ \iota >$$

où

- $< S,\ T,\ \alpha,\ \beta >$ est un système de transition simple sur S,
- $\gamma : S \rightarrow \wp(E)$ est la fonction de paramétrage des états qui associe à chaque état un sous-ensemble des paramètres d'états,
- $\iota : T \rightarrow \wp(F)$ est la fonction de paramétrage des transitions qui associe à chaque transition un sous-ensemble des paramètres de transition.

Remarque 2 Dans (1), on définit les systèmes paramétrés par E et F sur S avec $E = \{X_1,\ X_2, \ldots,\ X_n\}$ et $F = \{Y_1,\ Y_2, \ldots,\ Y_m\}$ comme des structures

$$\mathbf{A} =< S,\ T,\ \alpha,\ \beta,\ S_{X_1},\ \ldots, S_{X_n},\ T_{Y_1},\ \ldots, T_{Y_m} >$$

où

- $< S,\ T,\ \alpha,\ \beta >$ est un système de transition simple sur S,
- $\forall i, S_{X_i} \subseteq S$ est l'ensemble des états de l'automate ayant la propriété X_i,
- $\forall j : T_{Y_j} \subseteq T$ est l'ensemble des transitions de l'automate ayant la propriété Y_j.

  Ces deux notions de systèmes paramétrés sont équivalentes. En effet, il suffit de poser

  pour tout $X \in E,\ X \in \gamma(s)$ *ssi* $s \in S_X$, *et* *pour tout* $Y \in F,\ Y \in \gamma(s)$ *ssi* $s \in S_Y$,

  pour réaliser cette équivalence. L'avantage de la première présentation tient à ce qu'elle mène directement à la notion de structure de Kripke qui est la structure généralement

utilisée pour modéliser les systèmes intervenant en vérification de modèles. Dans une structure de Kripke, il n'y a pas de paramètres sur l'ensemble des transitions. C'est-à-dire que F est vide. On omet donc son existence.

Définition 15 Soit S ensemble et E un ensemble fini. Une *structure de Kripke* K sur S et E est la donnée d'un sextuplet

$$\mathbf{K} = < S,\ T,\ E,\ \alpha,\ \beta,\ \gamma >$$

où

- $< S,\ T,\ \alpha,\ \beta >$ est un système de transition simple sur S tel que l'image de α soit S en entier ($S \subseteq \alpha(T)$),
- E est un ensemble de paramètres d'états,
- $\gamma : S \rightarrow \wp(E)$ est la fonction de paramétrage des états qui associe à chaque état un sous-ensemble des paramètres d'états.

Dans (3), on trouve une autre définition de la notion de structure de Kripke, définie sur un ensemble PA de *propositions atomiques*, pour faire référence au caractère logique dans lequel elle est plongée. Ces «propositions» ne sont «vraies» qu'aux états auxquels elles sont associées via une fonction d'étiquetage $L : S \rightarrow \wp(PA)$ où S est l'ensembles des états de la structure. Plus précisément, soit PA un ensemble de propositions atomiques.

Définition 16 Une structure de Kripke $\mathbf{K}'$ sur PA est la donnée d'un quadruplet

$$\mathbf{K}' =< S,\ S_0,\ R,\ L >$$

où

- S est un ensemble fini d'états,
- $S_0 \subseteq S$ est l'ensemble des états *initiaux* de $\mathbf{K}'$,
- $R \subseteq S \times S$ est une relation totale sur S décrivant les transitions du système. Par *relation totale*, on désigne ici un sous-ensemble de $S \times S$ tel que pour tout élément $s \in S$ il existe un état $s' \in S$ tel que $(s, s') \in R$,
- $L : S \rightarrow \wp(PA)$ est une fonction qui associe à chaque état l'ensemble des propositions qui sont vraies dans cet état.

Il n'est pas difficile de voir que les deux notions de structure de Kripke dégagées plus haut sont équivalentes. En effet, l'ensemble S des états est décrit de la même manière dans les deux cas. La relation de transition donnée dans $\mathbf{K}$ par les fonctions α et γ est équivalente à la relation totale R de $\mathbf{K}'$ car on suppose avoir la relation d'inclusion $S \subseteq \alpha(T)$. Enfin, en ajoutant une propriété *initiale* à PA pour définir l'ensemble des paramètres d'états E de $\mathbf{K}$,

$$E = PA \cup \{initiale\}$$

puis en définissant γ par

$$\gamma : S \to \wp(E)$$

où

$$\forall x \in S,\ y \in \gamma(x) \Leftrightarrow ((y \in PA \text{ et } y \in L(x)) \text{ ou } (y = initiale \text{ et } x \in S_0))$$

on obtient clairement l'équivalence entre les deux fonctions γ et L, et, par conséquent, l'équivalence entre les deux notions de structures de Kripke précédentes. Dans la suite du mémoire, une structure de Kripke sera toujours entendue dans le sens de la définition 16.

Définition 17 Soit $\mathbf{A}$ un système de transition. Une *exécution* π dans $\mathbf{A}$ est une suite infinie d'états

$$\pi = s_0 s_1 s_2 s_3 \ldots$$

telle que pour tout entier $i \geq 0$, (s_i, s_{i+1}) est une transition dans l'ensemble des transitions T de $\mathbf{A}$. On dit aussi que π est un *chemin infini*, ou plus succinctement un chemin, faisant alors référence à la théorie des langages sur l'alphabet S ou à celle plus générale des monoïdes. On dénote par $\mu(\mathbf{A})$ (resp. $\mu(\mathbf{K})$) l'ensemble des exécutions possibles dans un système de transition $\mathbf{A}$ (resp. dans une structure de Kripke $\mathbf{K}$).

Définition 18 Soit une exécution $\pi = s_0 s_1 s_2 s_3 \ldots$ dans un système de transitions $\mathbf{A}$. Un *préfixe* de $\pi = s_0 s_1 s_2 s_3 \ldots$ est une suite finie $s'_0 s'_1 s'_2 s'_3 \ldots s'_k$ d'états de S telle que pour tout entier $i, 0 \leq i \leq k, s'_i = s_i$. Un *suffixe* d'une exécution $\pi = s_0 s_1 s_2 s_3 \ldots$ est une exécution $s'_k s'_{k+1} s'_{k+2} s'_{k+3} \ldots$ dans $\mathbf{A}$ où $k \geq 0$ et telle que pour tout entier $j \geq 0$, on a $s_{k+j} = s'_{k+j}$. On dénote ce suffixe par π^k. Une exécution peut aussi être vue comme une fonction de $\mathbf{N}$ dans S, $\pi : \mathbf{N} \to S$. Si $\pi = s_0 s_1 s_2 s_3 \ldots$ alors, pour tout entier, $i \geq 0$, on pose $\pi[i] = s_i$.

2.4 Les logiques temporelles

Lorsqu'on veut vérifier qu'un modèle informatique, qu'il soit donné sous forme de structure de Kripke ou autrement, possède une propriété particulière, on doit avoir à sa disposition une

manière de décrire la propriété voulue. En général, le formalisme utilisé pour décrire le modèle ne suffit pas à la tâche et il faut le compléter. C'est à ce moment qu'interviennent les logiques *temporelles* du système. On entend par là un langage, un formalisme supplémentaire à celui utilisé pour le modèle qui permette d'exprimer certaines propriétés que peuvent avoir ou ne pas avoir certains états et/ou exécutions du modèle en question. Ces logiques sont destinées à tenir compte de l'évolution des systèmes dans le temps (discret).

Il existe plusieurs logiques temporelles, adaptées à différents contextes, ayant des pouvoirs expressifs différents. Ces pouvoirs expressifs sont donnés par les opérateurs temporels qui constituent chaque logique. La logique que nous utiliserons et que nous généraliserons est appelée CTL, un acronyme pour «*Computation Tree Logic*». Nous l'introduisons ici via une logique plus puissante que CTL, appelée CTL^*. Elle est plus puissante dans le sens où CTL^* permet d'exprimer tout ce que CTL est en mesure de faire alors que la réciproque est fausse.

2.4.1 Les opérateurs temporels

Comme leurs noms l'indiquent, ces opérateurs décrivent des propriétés de structures qui tiennent compte du «temps» (passé, présent et futur) du système sous analyse. Par «temps», on entend ici en fait le déroulement discret (combinatoire) des événements produits par le système. Dit autrement, tous ces opérateurs expriment des propriétés que possédait et/ou possède et/ou possèdera le modèle du système à un moment ou un autre, c'est-à-dire lorsque le système est dans un état ou un autre, ou passe d'un état à un autre.

Ces opérateurs servent ainsi à exprimer certaines propriétés du *comportement* des systèmes. On suppose ici que ce comportement est «déployé», c'est-à-dire qu'il est donné sous forme d'arborescence orientée infinie représentant toutes les exécutions possibles du modèle du système. Dans ce contexte, les opérateurs indiqueront alors des propriétés sur les états (noeuds de l'arborescence) et sur les exécutions (chemins de l'arborescence) du modèle.

En général, la logique propositionnelle munie des quantificateurs universel et existentiel usuels, ce qu'on appelle aussi la logique du premier ordre, suffit à décrire de manière simple et efficace les propriétés des *états* du système. D'ailleurs, la logique du premier ordre pourrait aussi servir à décrire les propriétés des chemins. Mais cela se ferait alors au prix d'une écriture très lourde et, comme il est indiqué dans (14), au prix de redondances logiques qui ne conviennent pas à l'analyse des systèmes. Voici la liste des opérateurs temporels que nous examinerons dans

ce mémoire, donnés avec leur interprétation respective. Ils énoncent des propriétés de certaines exécutions (ou chemins) dans le déployé du comportement d'un système. Puisque ces exécutions sont en fait des suites d'états particulières, il est normal qu'on fasse référence *aussi* à des états du système. Cette énumération n'est pas exhaustive. On pourra en trouver plusieurs autres dans (13), par exemple, qui traitent notamment de propriétés *passées*.

Liste de certains opérateurs temporels

i. L'opérateur de chemin unaire **X**, (pour «ne**X**t»), indiquant qu'une propriété est vraie au prochain état d'une exécution donnée.

ii. L'opérateur de chemin unaire **F**, (pour «**F**utur»), indiquant qu'une propriété sera éventuellement vraie en un certain état d'une exécution donnée.

iii. L'opérateur de chemin unaire **G**, (pour «**G**énéral» ou «**G**lobalement»), indiquant qu'une propriété sera toujours vraie, c'est-à-dire qu'elle est vraie en tous les états d'une certaine exécution donnée.

iv. L'opérateur de chemin binaire **U**, (pour «**U**ntil»), indiquant qu'une première propriété sera vraie en tous les états d'un certain préfixe d'une exécution jusqu'à ce qu'une seconde propriété soit vraie en un certain état suivant immédiatement le préfixe.

v. L'opérateur de chemin binaire **R**, (pour «**R**elease»), indiquant que la seconde propriété sera vraie tout le long de l'exécution, jusqu'à ce que soit atteint le premier état où la première propriété est vraie. Cependant, il n'est pas exigé que la première propriété soit éventuellement vraie.

2.4.2 La logique CTL^*

Comme nous l'avons déjà mentionné, la logique CTL^* est essentiellement un *langage* permettant de décrire certaines propriétés du déployé d'un système codé sous forme de structure de Kripke $K =< S, S_0, R, L >$. Outre l'ensemble S donné par K, ce langage comporte aussi

1. des constantes logiques

$$\mathbf{1}\ (=\mathbf{vrai}),\ \mathbf{0}\ (=\mathbf{faux})$$

2. des connecteurs logiques,

$$\sim,\ \wedge,\ \vee$$

3. des quantificateurs logiques,

$$\mathbf{A},\ \mathbf{E}$$

4. des quantificateurs temporels,

$$\mathbf{X},\ \mathbf{F},\ \mathbf{G},\ \mathbf{U},\ \mathbf{R}$$

En logique, une propriété est une *formule bien formée*. On entend par là une formule qui a du sens syntaxiquement parlant. En général, on décrit l'ensemble de ces formules récursivement. La logique CTL^* est formée de deux classes distinctes de formules: celles qui concernent les états du système (Φ) et celles qui concernent les exécutions (Υ).

Définition 19 Soit PA un ensemble fini de propositions atomiques. L'ensemble $\Phi(PA)$ des *formules d'état* sur PA est défini récursivement de la manière suivante:

- $\mathbf{1} \in \Phi(PA),\ \mathbf{0} \in \Phi(PA)$,
- $\forall x \in PA,\ x \in \Phi(PA)$,
- $\forall f \in \Phi(PA),\ (\sim f) \in \Phi(PA)$,
- $\forall f, g \in \Phi(PA),\ (f\ \wedge\ g) \in \Phi(PA)$,
- $\forall f, g \in \Phi(PA),\ (f\ \vee\ g) \in \Phi(PA)$,
- $\forall f \in \Upsilon(PA),\ (\mathbf{A}f) \in \Phi(PA)$,
- $\forall f \in \Upsilon(PA),\ (\mathbf{E}f) \in \Phi(PA)$.

Il n'y a pas d'autres formules d'état que celles obtenues précédemment. Remarquons que les quantificateurs **A** et **E** prennent comme argument des formules de chemins.

L'interprétation *intuitive*, c'est-à-dire qui n'est liée à aucun système ou aucune représentation concrète particulier, de ces formules est le sens usuel des connecteurs et quantificateurs que l'on retrouve en logique du premier ordre, à savoir:

i. La formule $\sim f$ indique la *négation* de f et signifie qu'elle n'est vraie que si, et seulement si, la formule f est fausse.

ii. La formule $f \wedge g$ indique la *conjonction* de f et g et signifie qu'elle n'est vraie que si, et seulement si, les formules f et g sont vraies.

iii. La formule $f \vee g$ indique la *disjonction* de f et g et signifie qu'elle n'est vraie que si, et seulement si, les formules f ou g sont vraies.

iv. La formule $\mathbf{A}f$ indique une *quantification* logique sur un ensemble de chemins ayant une origine commune. Elle n'est n'est vraie que si, et seulement si, la formule f est vraie sur tous les chemins de l'ensemble donné.

v. La formule $\mathbf{E}f$ indique une *quantification* logique sur un ensemble de chemins ayant une origine commune. Elle n'est vraie que si, et seulement si, la formule f est vraie pour au moins un des chemins de l'ensemble donné.

Définition 20 Soit PA un ensemble fini de propositions atomiques. L'ensemble $\Upsilon(PA)$ des *formules de chemins* sur PA est défini de la manière suivante:

- $\mathbf{1} \in \Upsilon(PA),\ \mathbf{0} \in \Upsilon(PA),$
- $\forall f \in \Phi(PA),\ f \in \Upsilon(PA),$
- $\forall f \in \Upsilon(PA),\ (\sim f) \in \Upsilon(PA),$
- $\forall f, g \in \Upsilon(PA),\ (f\ \wedge\ g) \in \Upsilon(PA),$
- $\forall f, g \in \Upsilon(PA),\ (f\ \vee\ g) \in \Upsilon(PA),$
- $\forall f \in \Upsilon(PA),\ (\mathbf{X}f) \in \Upsilon(PA),$
- $\forall f \in \Upsilon(PA),\ (\mathbf{F}f) \in \Upsilon(PA),$
- $\forall f \in \Upsilon(PA),\ (\mathbf{G}f) \in \Upsilon(PA),$
- $\forall f, g \in \Upsilon(PA),\ (f\mathbf{U}g) \in \Upsilon(PA),$
- $\forall f, g \in \Upsilon(PA),\ (f\mathbf{R}g) \in \Upsilon(PA),$

Il n'y a pas d'autres formules de chemin que celles obtenues précédemment. Remarquons que les formules d'état sont aussi, par définition, des formules de chemin.

Il est à noter que les formules bien formées n'ont qu'un caractère syntaxique. Le sens intuitif de ces formules a été donné plus haut lorsque nous avons introduit les opérateurs temporels et que nous les avons interprétés dans le cadre d'un déployé abstrait, en termes d'états et d'exécutions d'un système. Pour leur donner un sens précis et rigoureux, on doit aussi disposer

des structures nécessaires permettant de *calculer* et *vérifier* les propriétés voulues c'est-à-dire d'un système d'*axiomes* donné sous une forme ou une autre, permettant de valider des énoncés. Dans le contexte que nous développons ici, les axiomes sont donnés par une structure de Kripke $K =< S,\ S_0,\ R,\ L >$ sur un ensemble PA de propositions atomiques, via l'ensemble PA. On peut alors définir la *sémantique* du système récursivement sous la forme d'une relation de satisfaction dénotée «$\models$».

Définition 21 Soient $K =< S,\ S_0,\ R,\ L >$ une structure de Kripke sur un ensemble PA de propositions atomiques, s un état de S, $\pi = s_0 s_1 s_2 \ldots$ une exécution dans K où $s_0 \in S_0$. La relation de satisfaction $\models$ de CTL* est définie ainsi:

1. $\forall p \in PA, \quad K, s \models p \underset{\text{def}}{\Leftrightarrow} p \in L(s)$
2. $\forall f \in \Phi(PA), \quad K, s \models \sim f \underset{\text{def}}{\Leftrightarrow} K, s \not\models f$
3. $\forall f_1, f_2 \in \Phi(PA), \quad K, s \models f_1 \vee f_2 \underset{\text{def}}{\Leftrightarrow} K, s \models f_1$ ou $K, s \models f_2$.
4. $\forall f_1, f_2 \in \Phi(PA), \quad K, s \models f_1 \wedge f_2 \underset{\text{def}}{\Leftrightarrow} K, s \models f_1$ et $K, s \models f_2$.
5. $\forall g \in \Upsilon(PA), \quad K, s \models \mathbf{E}g \underset{\text{def}}{\Leftrightarrow} \exists \pi$ tel que $\pi[0] = s$ et $K, \pi \models g$.
6. $\forall g \in \Upsilon(PA), \quad K, s \models \mathbf{A}g \underset{\text{def}}{\Leftrightarrow} \forall \pi$ tel que $\pi[0] = s$ on a $K, \pi \models g$.
7. $\forall f \in \Phi(PA), \quad K, \pi \models f \underset{\text{def}}{\Leftrightarrow} K, \pi[0] \models f$.
8. $\forall g \in \Upsilon(PA), \quad K, \pi \models \sim g \underset{\text{def}}{\Leftrightarrow} K, \pi[0] \not\models g$
9. $\forall g_1, g_2 \in \Upsilon(PA), \quad K, \pi \models g_1 \wedge g_2 \underset{\text{def}}{\Leftrightarrow} K, \pi \models g_1$ et $K, \pi \models g_2$.
10. $\forall g_1, g_2 \in \Upsilon(PA), \quad K, \pi \models g_1 \vee g_2 \underset{\text{def}}{\Leftrightarrow} K, \pi \models g_1$ ou $K, \pi \models g_2$.
11. $\forall g \in \Upsilon(PA), \quad K, \pi \models \mathbf{X}g \underset{\text{def}}{\Leftrightarrow} K, \pi^1 \models g$.
12. $\forall g \in \Upsilon(PA), \quad K, \pi \models \mathbf{F}g \underset{\text{def}}{\Leftrightarrow} \exists j \geq 0\ t.q.\ K, \pi^j \models g$.
13. $\forall g \in \Upsilon(PA), \quad K, \pi \models \mathbf{G}g \underset{\text{def}}{\Leftrightarrow} \forall k \geq 0,\ K, \pi^k \models g$.
14. $\forall g_1, g_2 \in \Upsilon(PA), \quad K, \pi \models g_1 \mathbf{U} g_2 \underset{\text{def}}{\Leftrightarrow} \exists k \geq 0\ t.q.\ K, \pi^k \models g_2$ et $\forall j,\ 0 \leq j < k,\ K, \pi^j \models g_1$.
15. $\forall g_1, g_2 \in \Upsilon(PA), \quad K, \pi \models g_1 \mathbf{R} g_2 \underset{\text{def}}{\Leftrightarrow} \forall j \geq 0$ si $\forall i, 0 \leq i < j,\ K, \pi^i \not\models g_1$ alors $K, \pi^j \models g_2$.

Notons que les relations d'insatisfactions $K, s \not\models f$ et $K, \pi \not\models g$ peuvent être obtenues en calculant respectivement les valeurs $\sim (K, s \models f)$ et $\sim (K, \pi \models g)$ récursivement à partir des définitions précédentes.

La relation de satisfaction «$\models$» peut être revue dans un contexte plus facile à généraliser que celui qui vient d'être présenté. En utilisant la notion de fonction ou *valeur de vérité* d'une formule telle qu'elle est dégagée, par exemple, dans (8, p. 36) au sujet du calcul propositionnel multivalent, on peut considérer que cette relation est satisfaite seulement lorsqu'elle est vraie ou encore lorsque sa valeur de vérité est égale à un. Sinon, elle est fausse ou encore sa *valeur de vérité* est nulle. En dénotant respectivement par

$$\mathbf{e} : S \times \Phi(PA) \rightarrow \{0, 1\}, \quad (s, f) \mapsto \mathbf{e}(s, f) = \mathbf{e}_s(f)$$

et, avec abus,

$$\mathbf{e} : \mu(K) \times \Upsilon(PA) \rightarrow \{0, 1\}, \quad (\pi, g) \mapsto \mathbf{e}(\pi, g) = \mathbf{e}_\pi(g)$$

les fonctions permettant d'évaluer la valeur de vérité d'une formule d'états et, respectivement, de chemins relativement à une structure de Kripke fixe K, on obtient alors la liste des évaluations correspondantes donnée dans la définition suivante.

Définition 22 Soient $K =< S,\ S_0,\ R,\ L >$ une structure de Kripke sur un ensemble PA de propositions atomiques, s un état de S, $\pi = s_0 s_1 s_2 \ldots$ une exécution dans K. La fonction de vérité $\mathbf{e}$ sur K est définie ainsi:

1. $\forall p \in PA, \quad \mathbf{e}_s(p) = 1 \underset{\text{def}}{\Leftrightarrow} p \in L(s).$
2. $\forall f \in \Phi(PA), \quad \mathbf{e}_s(\sim f) = 1 - \mathbf{e}_s(f).$
3. $\forall f_1, f_2 \in \Phi(PA), \quad \mathbf{e}_s(f_1 \vee f_2) = \mathrm{Max}(\mathbf{e}_s(f_1), \mathbf{e}_s(f_2)).$
4. $\forall f_1, f_2 \in \Phi(PA), \quad \mathbf{e}_s(f_1 \wedge f_2) = \mathrm{Min}(\mathbf{e}_s(f_1), \mathbf{e}_s(f_2)).$
5. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_s(\mathbf{E}g) = \max_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(g).$
6. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_s(\mathbf{A}g) = \min_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(g).$
7. $\forall f \in \Phi(PA), \quad \mathbf{e}_\pi(f) = \mathrm{Min}(\chi(\pi[0] = s), \mathbf{e}_s(f)).$
8. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\sim g) = 1 - \mathbf{e}_\pi(g).$

9. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \wedge g_2) = \mathrm{Min}(\mathbf{e}_\pi(g_1), \mathbf{e}_\pi(g_2))$.

10. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \vee g_2) = \mathrm{Max}(\mathbf{e}_\pi(g_1), \mathbf{e}_\pi(g_2))$.

11. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{X}g) = \mathbf{e}_{\pi^1}(g)$.

12. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{F}g) = \max_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g)$.

13. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{G}g) = \min_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g)$.

14. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \mathbf{U} g_2) = \max_{k \in \mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \leq j < k}\ \mathbf{e}_{\pi^j}(g_1)))$.

15. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \mathbf{R} g_2) = \min_{k \in \mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g_2), \max_{0 \leq j < k}\ \mathbf{e}_{\pi^j}(g_1)))$.

où la fonction $\chi : \mathbf{B} \to \mathbf{B}'$ est définie par

$$\chi(x) = \begin{cases} 0, & \text{si } x = \mathbf{faux}, \\ 1, & \text{si } x = \mathbf{vrai}. \end{cases}$$

Proposition 4 (Équivalences) *Les définitions 21 et 22 sont équivalentes, dans le sens où pour toute structure de Kripke $K =< S,\ S_0,\ R,\ L >$ sur un ensemble fini AP, toute formule d'état $f \in \Phi(AP)$ et toute formule de chemin $g \in \Upsilon(PA)$ on a, pour tout élément s de S et tout chemin π de $\mu(K)$*

$$K, s \models f \Leftrightarrow \mathbf{e}_s(f) = 1$$

et

$$K, \pi \models g \Leftrightarrow \mathbf{e}_\pi(g) = 1.$$

Démonstration. Remarquons d'abord qu'à toute formule d'état $f \in \Phi(AP)$ et toute formule de chemin $g \in \Upsilon(PA)$ on peut associer sa *longueur* consistant en le nombre de fois que l'on a appliqué une des règles dans les définitions 19 et 20 pour les obtenir. Cette longueur est bien définie car il n'y a pas de règles d'élimination dans les dernières définitions. De plus, toute formule ne comporte qu'un nombre fini de décompositions via ces mêmes définitions. On peut donc procéder sans crainte par récurrence sur la longueur des formules pour démontrer ce théorème.

Si la formule p est de longueur 1, c'est-à-dire si $p \in PA$ alors on a

$$\forall p \in PA,\ K, s \models p \Leftrightarrow p \in L(s) \Leftrightarrow \mathbf{e}_s(p) = 1.$$

et

$$\forall p \in PA,\ K, \pi \models p \Leftrightarrow \pi[0] = s \text{ et } K, s \models p \Leftrightarrow \mathrm{Min}(\chi(\pi[0] = s), \mathbf{e}_s(p)) = 1,$$

et la proposition est vraie. Supposons qu'elle le soit pour les formules de longueur n *ou moins.* Il suffit alors de vérifier que chacun des 14 points (sauf le premier) de la définition 21 est équivalent à celui de la définition 22 en supposant que les formules évaluées sont de longueur $n+1$. Le résultat découlera alors de l'hypothèse de récurrence car toutes ces formules sont issues d'applications d'opérateurs sur des formules de longueur strictement plus petites que $n+1$.

Les point 1 à 10 sont évidents et découlent immédiatement des propriétés élémentaires des fonctions Max et Min. Pour les points 11, 12 et 14, on a

$$\begin{aligned}
\mathbf{e}_\pi(\mathbf{X}g) = 1 &\Leftrightarrow \mathbf{e}_{\pi^1}(g) = 1 \\
&\Leftrightarrow \mathrm{Min}(\chi(\pi^1[0] = s), \mathbf{e}_s(g)) = 1 \\
&\Leftrightarrow (\chi(\pi^1[0] = s) \wedge \mathbf{e}_s(g)) = 1 \\
&\Leftrightarrow K, \pi^1 \models (g) \\
&\Leftrightarrow K, \pi \models \mathbf{X}(g)
\end{aligned}$$

$$\begin{aligned}
\mathbf{e}_\pi(\mathbf{F}g) = 1 &\Leftrightarrow \max_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g) = 1 \\
&\Leftrightarrow \exists j_0 \in \mathbf{N}\ \mathbf{e}_{\pi^{j_0}}(g) = 1 \\
&\Leftrightarrow \exists j_0 \in \mathbf{N}\ K, \pi^{j_0} \models g \\
&\Leftrightarrow K, \pi \models \mathbf{F}g
\end{aligned}$$

$$\begin{aligned}
\mathbf{e}_\pi(f\mathbf{U}g) = 1 &\Leftrightarrow \max_{k \in \mathbf{N}}(\mathrm{Min}(\mathbf{e}_{\pi^k}(g), \min_{0 \le j < k} \mathbf{e}_{\pi^j}(f))) = 1 \\
&\Leftrightarrow \exists k \in \mathbf{N}, \mathrm{Min}(\mathbf{e}_{\pi^k}(g), \min_{0 \le j < k} \mathbf{e}_{\pi^j}(f)) = 1 \\
&\Leftrightarrow \exists k \in \mathbf{N}, \mathbf{e}_{\pi^k}(g) = 1 \ \wedge \min_{0 \le j < k} \mathbf{e}_{\pi^j}(f) = 1 \\
&\Leftrightarrow \exists k \in \mathbf{N}, K, \pi^k \models g \ \wedge\ \forall j, 0 \le j < k,\ \mathbf{e}_{\pi^j}(f) = 1 \\
&\Leftrightarrow K, \pi \models (f\mathbf{U}g).
\end{aligned}$$

Les points 13 et 15 se traitent de la même façon que les points 12 et 14, respectivement.

Proposition 5 (Dualité) *Le couple des opérations* $(\vee, \wedge)$*, le couple de quantificateurs* $(\mathbf{A}, \mathbf{E})$ *de même que les couples d'opérateurs temporels* $(\mathbf{F}, \mathbf{G})$ *et* $(\mathbf{U}, \mathbf{R})$ *sont duaux l'un de l'autre, dans le sens où pour toute structure de Kripke* K *et toutes formules* f, g *dans* $\Upsilon(K)$ *on a:*

1) $f \vee g = \sim (\sim f \wedge \sim g)$ *et* $f \wedge g = \sim (\sim f \vee \sim g)$,

2) $\mathbf{A}f = \sim \mathbf{E} \sim f$ *et* $\mathbf{E}f = \sim \mathbf{A} \sim f$,

3) $\mathbf{F}g = \sim \mathbf{G} \sim g$ *et* $\mathbf{G}g = \sim \mathbf{F} \sim g$,

4) $(f\mathbf{U}g) = \sim (\sim f)\mathbf{R}(\sim g)$ *et* $(f\mathbf{R}g) = \sim (\sim f)\mathbf{U}(\sim g)$.

Démonstration. Nous ne démontrons que la première partie de chaque assertion. La deuxième se fait de manière similaire, par dualité des fonctions Min et Max pour $[0,1]$. Pour tout chemin π dans $\mu(K)$ et tout état s dans l'ensemble des états de K, on a:

$$\begin{aligned}
\mathbf{e}_s(f \vee g) &= 1 - (1 - \mathbf{e}_s(f \vee g)) \\
&= 1 - (1 - \mathrm{Max}(\mathbf{e}_s(f),\ \mathbf{e}_s(g))) \\
&= 1 - (\mathrm{Min}(1 - \mathbf{e}_s(f),\ 1 - \mathbf{e}_s(g))) \\
&= 1 - (\mathrm{Min}(\mathbf{e}_s \sim (f),\ \mathbf{e}_s \sim (g))) \\
&= 1 - \mathbf{e}_s(\sim f \wedge \sim g) \\
&= \mathbf{e}_s(\sim (\sim f \wedge \sim g))
\end{aligned}$$

$$\begin{aligned}
\mathbf{e}_s(\mathbf{A}g) &= 1 - (1 - \mathbf{e}_s(\mathbf{A}g)) \\
&= 1 - (1 - \max_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(g)) \\
&= 1 - \min_{\pi \in \mu(K), \pi[0]=s} (1 - \mathbf{e}_\pi(g)) \\
&= 1 - \min_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(\sim g) \\
&= 1 - \mathbf{e}_s(\mathbf{E} \sim g) \\
&= \mathbf{e}_s(\sim \mathbf{E} \sim g)
\end{aligned}$$

$$\begin{aligned}
\mathbf{e}_\pi(\mathbf{F}g) &= 1 - (1 - \mathbf{e}_\pi(\mathbf{F}g)) \\
&= 1 - (1 - \max_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g))
\end{aligned}$$

$$
\begin{aligned}
&= 1 - \min_{j \in \mathbf{N}}(1 - \mathbf{e}_{\pi^j}(g)) \\
&= 1 - \min_{j \in \mathbf{N}}(\mathbf{e}_{\pi^j}(\sim g)) \\
&= 1 - \mathbf{e}_{\pi}(\mathbf{G}(\sim g)) \\
&= \mathbf{e}_{\pi}(\sim \mathbf{G} \sim g)
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{e}_{\pi}(g_1 \mathbf{U} g_2) &= \max_{k \in \mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \le j < k}\ \mathbf{e}_{\pi^j}(g_1)))) \\
&= 1 - (1 - \max_{k \in \mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \le j < k}\ \mathbf{e}_{\pi^j}(g_1))) \\
&= 1 - \min_{k \in \mathbf{N}}(1 - \mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \le j < k}\ \mathbf{e}_{\pi^j}(g_1))) \\
&= 1 - \min_{k \in \mathbf{N}}(\mathrm{Max}(1 -\ \mathbf{e}_{\pi^k}(g_2), 1 - \min_{0 \le j < k}\ \mathbf{e}_{\pi^j}(g_1))) \\
&= 1 - \min_{k \in \mathbf{N}}(\mathrm{Max}(1 -\ \mathbf{e}_{\pi^k}(g_2), \max_{0 \le j < k}\ 1 - \mathbf{e}_{\pi^j}(g_1))) \\
&= 1 - \min_{k \in \mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(\sim g_2), \max_{0 \le j < k}\ \mathbf{e}_{\pi^j}(\sim g_1))) \\
&= 1 - (\ \mathbf{e}_{\pi}(\sim g_1 \mathbf{R} \sim g_2)) \\
&= \ \mathbf{e}_{\pi} \sim (\sim g_1 \mathbf{R} \sim g_2)
\end{aligned}
$$

Proposition 6 (Propriétés) *Pour toute structure de Kripke K et toute formule f dans $\Upsilon(K)$ on a*

1. $\mathbf{F}f \equiv$ **true U** f

2. $\mathbf{A}[f \wedge g] = \mathbf{A}[f] \wedge \mathbf{A}[g]$

3. $\mathbf{E}[f \vee g] = \mathbf{E}[f] \vee \mathbf{E}[g]$

Démonstration. Nous démontrons seulement la première propriété. Puisque **true** et f sont toutes deux des formules de chemins, on a, pour tout chemin π dans $\mu(K)$,

$$
\begin{aligned}
\mathbf{e}_{\pi}(\mathbf{true\ U}\ f) &= \max_{k \in \mathbf{N}}(\mathrm{Min}(\mathbf{e}_{\pi^k}(f), \min_{0 \le j < k}\ \mathbf{e}_{\pi^j} 1)) \\
&= \max_{k \in \mathbf{N}}(\mathrm{Min}(\mathbf{e}_{\pi^k}(f), 1)) \\
&= \max_{k \in \mathbf{N}}(\mathbf{e}_{\pi^k}(f)) \\
&= \mathbf{e}_{\pi}(\mathbf{F}f)
\end{aligned}
$$

Les points 2 et 3 découlent directement de la commutativité et de l'associativité des opérations binaires Max et Min.

2.4.3 La logique CTL

La logique que nous voulons étudier en logique floue est la logique CTL. Vue comme un sous-ensemble de CTL^*, on n'y considère seulement que des paires ordonnées d'opérateurs de la forme

$$\mathbf{QT}$$

où $\mathbf{Q} \in \{\mathbf{A}, \mathbf{E}\}$ est un quantificateur de chemin et $\mathbf{T} \in \{\mathbf{X}, \mathbf{F}, \mathbf{G}, \mathbf{U}, \mathbf{R}\}$ est un opérateur temporel. Il s'agit bel et bien d'une composition d'opérateurs $\mathbf{Q} \circ \mathbf{T}$ au sens de la composition usuelle des fonctions agissant sur des ensembles de formules. On la dénote $\mathbf{QT}$ car il n'y a pas de confusion possible.

Cette composition est bien définie. D'une part, le résultat d'une opération quelconque, définie dans le cadre de ce mémoire, sur une ou plusieurs formules de chemins retourne toujours une formule de chemins

$$\mathbf{T} : \Upsilon(E) \to \Upsilon(E)$$

ou

$$\mathbf{T} : \Upsilon(E) \times \Upsilon(E) \to \Upsilon(E).$$

D'autre part, les quantificateurs **A**, **E** prennent tous deux des formules de chemins comme arguments et produisent une formule d'états

$$\mathbf{T} : \Upsilon(E) \to \Phi(E),$$

de sorte que la composée décrite plus haut est bien définie. Ajoutons que la composition de deux de ces paires

$$(\mathbf{Q_1T_1})(\mathbf{Q_2T_2})$$

est aussi bien définie car une formule d'états est aussi une formule de chemins. Ainsi, pour toute formule de chemins f la formule $(\mathbf{Q_2T_2})(f)$ est aussi une formule de chemins.

Pour évaluer le composé $(\mathbf{Q} \circ \mathbf{T})$ on définit, pour toute formule de chemin f,

$$\mathbf{e}_s(\mathbf{Q} \circ \mathbf{T})(f) = \mathbf{e}_s(\mathbf{Q}(\mathbf{T}(\mathbf{f}))).$$

La logique CTL compte 10 opérateurs temporels de base:

- **AX**, **EX**
- **AF**, **EF**

- **AG, EG**
- **AU, EU**
- **AR, ER**

Il y a une certaine redondance dans ces opérateurs. En effet, il suffit de trois d'entre eux pour combler le pouvoir expressif de CTL, à savoir **EX**, **EG** et **EF**. Les sept autres s'expriment tous en fonction de ces derniers et de connecteurs de la logique propositionnelle, comme le démontre le théorème suivant.

Théorème 4 *Soient f et g des formules de chemins dans CTL. Alors on a*

1. $\mathbf{AX}f = \sim \mathbf{EX} \sim f$.
2. $\mathbf{EF}f = \mathbf{E}\ [\text{Vrai}\ \mathbf{U}\ f]$.
3. $\mathbf{AG}f = \sim \mathbf{EF} \sim f$.
4. $\mathbf{AF}f = \sim \mathbf{EG} \sim f$.
5. $\mathbf{A}[f\ \mathbf{R}\ g] = \sim \mathbf{E}[\sim f\ \mathbf{U}\ \sim g]$.
6. $\mathbf{A}[f\ \mathbf{U}\ g] = \sim \mathbf{E}[\sim g\ \mathbf{U}\ (\sim f \wedge \sim g)] \wedge \sim \mathbf{EG} \sim g$.

Démonstration. Soit K une structure de Kripke, $s \in S$ un état de K et $\mu(K)$ l'ensemble des exécutions de K.

1.

$$\begin{aligned}
K, s \models \mathbf{AX}f &\Leftrightarrow \forall \pi \in \mu(K) t.q. \pi[0] = s, K, \pi \models \mathbf{X}f \\
&\Leftrightarrow \forall \pi \in \mu(K) t.q. \pi[0] = s, K, \pi^1 \models f \\
&\Leftrightarrow \sim\sim \forall \pi \in \mu(K) t.q. \pi[0] = s, K, \pi^1 \models f \\
&\Leftrightarrow \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, \sim K, \pi^1 \models f \\
&\Leftrightarrow \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, K, \pi^1 \models \sim f \\
&\Leftrightarrow \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, K, \pi \models \mathbf{X} \sim f \\
&\Leftrightarrow K, s \models \sim \mathbf{EX} \sim f
\end{aligned}$$

2.

$$K, s \models \mathbf{EF}f \quad \Leftrightarrow \quad \exists \pi \in \mu(K) t.q.\ \pi[0] = s, K, \pi \models \mathbf{F}f$$

$$
\begin{aligned}
&\Leftrightarrow \quad \exists \pi \in \mu(K) t.q.\ \pi[0] = s, \exists n \geq 0,\ K, \pi^n \models f \\
&\Leftrightarrow \quad \exists \pi \in \mu(K) t.q.\ \pi[0] = s, \exists n \geq 0,\ K, \pi^n \models f \\
&\qquad \wedge \forall j, 0 \leq j\ k,\ K, \pi^j \models \mathbf{Vrai} \\
&\Leftrightarrow \quad \exists \pi \in \mu(K) t.q.\ \pi[0] = s,\ K, \pi \models\ [\mathbf{Vrai}\ \mathbf{U}\ f] \\
&\Leftrightarrow \quad K, s \models \mathbf{E}\ [\mathbf{Vrai}\ \mathbf{U}\ f]
\end{aligned}
$$

3.

$$
\begin{aligned}
K, s \models \mathbf{AG}f \quad &\Leftrightarrow \quad \forall \pi \in \mu(K)\ t.q. \pi[0] = s, K, \pi \models \mathbf{G}f \\
&\Leftrightarrow \quad \forall \pi \in \mu(K) t.q. \pi[0] = s, \forall n \geq 0, \pi^n \models f \\
&\Leftrightarrow \quad \sim\sim \forall \pi \in \mu(K) t.q. \pi[0] = s, \forall n \geq 0, \pi^n \models f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, \exists n \geq 0, \sim \pi^n \models f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, \exists n \geq 0, \pi^n \models\sim f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, K, \pi \models \mathbf{F} \sim f \\
&\Leftrightarrow \quad K, s \models\sim \mathbf{EF}f
\end{aligned}
$$

4.

$$
\begin{aligned}
K, s \models \mathbf{AF}f \quad &\Leftrightarrow \quad \forall \pi \in \mu(K) t.q. \pi[0] = s, K, \pi \models \mathbf{F}f \\
&\Leftrightarrow \quad \forall \pi \in \mu(K) t.q. \pi[0] = s, \exists n \geq 0, \pi^n \models f \\
&\Leftrightarrow \quad \sim\sim \forall \pi \in \mu(K) t.q. \pi[0] = s, \exists n \geq 0, \pi^n \models f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, \forall n \geq 0, \sim \pi^n \models f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, \forall n \geq 0, \pi^n \models\sim f \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K) t.q. \pi[0] = s, K, \pi \models \mathbf{G} \sim f \\
&\Leftrightarrow \quad K, s \models\sim \mathbf{EG}f
\end{aligned}
$$

5.

$$
\begin{aligned}
K, s \models \mathbf{A}[f\mathbf{R}g] \quad &\Leftrightarrow \quad \forall \pi \in \mu(K)\ t.q. \pi[0] = s,\ K, \pi \models [f\mathbf{R}g] \\
&\Leftrightarrow \quad \forall \pi \in \mu(K)\ t.q. \pi[0] = s,\ \forall j \geq 0,\ ((\forall i,\ t.q.\ i < j,\ K, \pi^i \not\models f) \Rightarrow K, \pi^j \models g) \\
&\Leftrightarrow \quad \forall \pi \in \mu(K)\ t.q. \pi[0] = s,\ \forall j \geq 0,\ ((\exists i,\ t.q.\ i < j,\ K, \pi^i \models f) \vee K, \pi^j \models g) \\
&\Leftrightarrow \quad \sim\sim \forall \pi \in \mu(K)\ t.q. \pi[0] = s,\ \forall j \geq 0,\ ((\exists i,\ t.q.\ i < j,\ K, \pi^i \models f) \vee K, \pi^j \models g) \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K)\ t.q. \pi[0] = s,\ \exists j \geq 0,\ ((\forall i,\ t.q.\ i < j,\ K, \pi^i \not\models f) \wedge K, \pi^j \not\models g) \\
&\Leftrightarrow \quad \sim \exists \pi \in \mu(K)\ t.q. \pi[0] = s,\ \exists j \geq 0,\ ((\forall i,\ t.q.\ i < j,\ K, \pi^i \models\sim f) \wedge K, \pi^j \models\sim g) \\
&\Leftrightarrow \quad K, s \models\sim \mathbf{E}[\sim f\mathbf{R} \sim g]
\end{aligned}
$$

6. La preuve du dernier item est moins directe que les précédentes. Notons d'abord par le numéro 4. de ce théorème que l'on a

$$\sim \mathbf{EG} \sim f = \mathbf{AF} \sim f.$$

Par le numéro 5. de ce théorème, on a aussi

$$\begin{aligned} \sim \mathbf{E}[\sim g \ \mathbf{U} \ (\sim f \wedge \sim g)] \quad &= \sim \mathbf{E}[\sim g \ \mathbf{U} \ \sim (f \vee g)] \\ &= \mathbf{A}[g \ \mathbf{R} \ (f \vee g)]. \end{aligned}$$

Ainsi, il suffit de montrer que $\mathbf{A}[f \ \mathbf{U} \ g] = \mathbf{A}[g \ \mathbf{R} \ (f \vee g)] \wedge \mathbf{AF}(g)$. Par la proposition 6, on a

$$\mathbf{A}([g \ \mathbf{R} \ (f \vee g)]) \wedge \mathbf{AF}(g) = \mathbf{A}([g \ \mathbf{R} \ (f \vee g)] \wedge \mathbf{F}g).$$

Le problème revient finalement à démontrer que

$$[f\mathbf{U}g] = [g \ \mathbf{R} \ (f \vee g)] \wedge \mathbf{F}g.$$

Supposons que l'on ait $K, \pi \models [f\mathbf{U}g]$. Alors, par définition, il existe un entier $n \geq 0$ tel que $K, \pi^n \models g$ et pour tout entier m tel que $0 \leq m < n$, on a $\pi^m \models f$. Prenons le plus petit entier n_0 tel que $K, \pi^{n_0} \models g$. Alors n_0 satisfait la condition $\mathbf{F}g$, c'est-à-dire que g est éventuellement satisfaite sur le chemin. De plus, en chaque indice $0 \leq i < n_0$ (s'il en existe), $K, \pi^i \models f$. Cela est équivalent à dire que pour tout entier $0 \leq j$, si, pour tout entier k tel que $0 \leq k < j$, on a $K, \pi^k \not\models g$ alors $K, \pi^j \models f \vee g$. En effet, puisque n_0 est le premier indice où g est satisfaite, si $j < n_0$ on a $K, \pi^j \models f \vee g$ entraîne que $K, \pi^j \models f$. La réciproque se démontre avec des arguments similaires.

Toute formule f de CTL peut être considérée comme un ensemble $f \in \wp(S)$ tel que

$$f \equiv \{s \in S | \quad K, s \models f\}.$$

Dans ce contexte, les opérateurs de base de CTL peuvent alors être vus comme les plus grands ou plus petits points fixes de certaines transformations de prédicats. Nous allons montrer que les couples $\mathbf{AF}f$, $\mathbf{EF}f$, $\mathbf{A}[f\mathbf{U}g]$, $\mathbf{E}[f\mathbf{U}g]$ (resp. $\mathbf{AG}f$, $\mathbf{EG}f$, $\mathbf{A}[f\mathbf{R}g]$, $\mathbf{E}[f\mathbf{R}g]$) sont les points fixes, dans l'ordre des transformations de prédicats $\tau_1, \tau_2, \tau_5, \tau_6$ (resp. $\tau_3, \tau_4, \tau_7, \tau_8$) définies comme suit.

Définition 23 Pour tout entier $1 \leq i \leq 8$, on pose

$$\tau_i : \wp(S) \to \wp(S)$$

où

1. $\tau_1(Z) = f \vee \mathbf{AX}Z$
2. $\tau_2(Z) = f \vee \mathbf{EX}Z$
3. $\tau_3(Z) = f \wedge \mathbf{AX}Z$
4. $\tau_4(Z) = f \wedge \mathbf{EX}Z$
5. $\tau_5(Z) = g \vee (f \wedge \mathbf{AX}Z)$
6. $\tau_6(Z) = g \vee (f \wedge \mathbf{EX}Z)$
7. $\tau_7(Z) = g \wedge (f \vee \mathbf{AX}Z)$
8. $\tau_8(Z) = g \wedge (f \vee \mathbf{EX}Z)$

L'interprétation de ces dernières formules se fait comme suit: il faut voir le membre droit de chacune comme un processus engendrant *récursivement* un sous-ensemble des états de S satisfaisant la condition présente. On a

1. $\tau_1(Z) = f \vee \mathbf{AX}Z$ indique que $s \in S$ est un élément de $\tau_1(Z)$ si et seulement si $K, s \models f$ **ou** que pour *toute* (**A** dans la formule) *transition* (**X** dans la formule) (s, s') ayant s comme origine, on a s' satisfaisant $K, s' \models Z$ c'est-à-dire que s' est dans Z.
2. $\tau_2(Z) = f \vee \mathbf{EX}Z$ indique que $s \in S$ est un élément de $\tau_2(Z)$ si et seulement si $K, s \models f$ **ou** qu'il existe (**E** dans la formule) une transition (**X** dans la formule) (s, s') ayant s comme origine, pour laquelle on a s' dans Z.
3. τ_3 a la même interprétation que τ_1 mis à part qu'il faille remplacer **ou** par **et**.
4. τ_4 a la même interprétation que τ_2 mis à part qu'il faille remplacer **ou** par **et**.
5. $\tau_5(Z) = g \vee (f \wedge \mathbf{AX}Z)$ indique que $s \in S$ est un élément de $\tau_5(Z)$ si et seulement si $K, s \models g$

   ou

$K, s \models f$ **et** pour toute (**A** dans la formule) transition (**X** dans la formule) (s, s') ayant s comme origine, on a s' satisfaisant $K, s' \models Z$ c'est-à-dire que s' est dans Z.

6. $\tau_6(Z) = g \vee (f \wedge \mathbf{EX}Z)$ indique que $s \in S$ est un élément de $\tau_6(Z)$ si et seulement si $K, s \models g$

   ou

   $K, s \models f$ **et** il existe (**E** dans la formule) une transition (**X** dans la formule) (s, s') ayant s comme origine, pour laquelle on a s' satisfaisant $K, s' \models Z$ c'est-à-dire que s' est dans Z.

7. τ_7 a la même interprétation que τ_5 mis à part qu'il faille remplacer **ou** par **et** et réciproquement.

8. τ_8 a la même interprétation que τ_6 mis à part qu'il faille remplacer **ou** par **et** et réciproquement.

Nous voulons montrer que les transformations précédentes engendrent les opérateurs de base de CTL, c'est-à-dire que ces derniers sont obtenus comme étant les plus petits ou les plus grands points fixes de ces transformations. Pour ce faire, on montre d'abord que les transformations sont monotones, que l'opérateur correspondant est bien un point fixe de la transformation et finalement que c'en est le plus petit ou le plus grand. Pour simplifier l'écriture, nous supposons la structure de Kripke K fixée d'ici jusqu'à la fin du chapitre permettant d'écrire $s \models f$ plutôt que $K, s \models f$.

Lemme 2 *Pour tout entier i, $1 \leq i \leq 8$, la fonction $\tau_i(Z)$ est monotone.*

Démonstration.

Supposons $P_1 \subseteq P_2 \subset S$. Il s'agit de montrer que, pour tout entier $i, 1 \leq i \leq 8$ on a $\tau_i(P_1) \subseteq \tau_i(P_2) \subseteq S$.

1. Pour tout $s \in \tau_1(P_1)$ on a $s \models f$ **ou** pour tout $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$ par définition de τ_1. Si $s \models f$ alors $s \in \tau_1(P_2)$, par définition de τ_1. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_1(P_2)$.

2. Pour tout $s \in \tau_1(P_1)$ on a $s \models f$ **ou** il existe $s' \in S$ tel que $(s, s') \in R$ pour lequel on a $s' \in P_1$, par définition de τ_2. Si $s \models f$ alors $s \in \tau_2(P_2)$, par définition de τ_2. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_2(P_2)$.

3. Pour tout $s \in \tau_3(P_1)$ on a $s \models f$ **et** pour tout $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$ par définition de τ_3. Puisque $P_1 \subseteq P_2$, on a $s' \in P_2$ ce qui entraîne que $s \in \tau_3(P_2)$.

4. Pour tout $s \in \tau_4(P_1)$ on a $s \models f$ **et** il existe $s' \in S$ tel que $(s, s') \in R$ pour lequel on a $s' \in P_1$, par définition de τ_4. Puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_4(P_2)$.

5. Pour tout $s \in \tau_5(P_1)$ on a:

   $s \models g$

   ou

   on a $s \models f$ **et** pour tout $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$ (par définition de τ_5).

   Si $s \models g$ alors $s \in \tau_5(P_2)$, par définition de τ_5. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_5(P_2)$.

6. Pour tout $s \in \tau_6(P_1)$ on a:

   $s \models g$

   ou

   on a $s \models f$ **et** il existe $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$, par définition de τ_6,

   Si $s \models g$ alors $s \in \tau_6(P_2)$, par définition de τ_5. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_6(P_2)$.

7. Pour tout $s \in \tau_7(P_1)$ on a:

   $s \models g$

   et

   on a $s \models f$ **ou** pour tout $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$ (par définition de τ_5).

   Si $s \models g$ alors $s \in \tau_7(P_2)$, par définition de τ_7. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_7(P_2)$.

8. Pour tout $s \in \tau_8(P_1)$ on a:

   $s \models g$

   et

   on a $s \models f$ **ou** pour tout $s' \in S$ tel que $(s, s') \in R$ on a $s' \in P_1$ (par définition de τ_7).

   Si $s \models g$ alors $s \in \tau_8(P_2)$, par définition de τ_8. Sinon, puisque $P_1 \subseteq P_2$ alors on a $s' \in P_2$ ce qui entraîne que $s \in \tau_8(P_2)$.

Lemme 3 *Pour tout entier $i \in \{1,2,5,6\}$ soit $\tau_i^{j_i}(\emptyset)$ la limite de la suite*

$$\emptyset \subseteq \tau_i(\emptyset) \subseteq \tau_i^2(\emptyset) \subseteq \cdots$$

et, pour tout entier $i \in \{3,4,7,8\}$ soit $\tau_i^{j_i}(S)$ la limite de la suite

$$S \supseteq \tau_i(S) \supseteq \tau_i^2(S) \supseteq \cdots$$

Alors pour tout s dans S,

1. *si s est dans $\tau_1^{j_1}(\emptyset)$ on a $s \models f$ ou, pour tout état $s' \in S$ tel que $(s,s') \in R$, on a $s' \in \tau^{j_1}(\emptyset)$.*

2. *si s est dans $\tau_2^{j_2}(\emptyset)$ on a $s \models f$ ou il existe un état $s' \in S$ tel que $(s,s') \in R$ et $s' \in \tau^{j_2}(\emptyset)$.*

3. *si s est dans $\tau_3^{j_3}(S)$ on as $\models f$ et, pour tout état $s' \in S$ tel que $(s,s') \in R$, on a $s' \in \tau^{j_3}(S)$.*

4. *si s est dans $\tau_4^{j_4}(S)$ on as $\models f$ et il existe un état $s' \in S$ tel que $(s,s') \in R$ on a $s' \in \tau^{j_4}(S)$.*

5. *si s est dans $\tau_5^{j_5}(\emptyset)$ on a $s \models g$ ou alors $s \models f$ et pour tout état $s' \in S$ tel que $(s,s') \in R$, on a $s' \in \tau^{j_5}(\emptyset)$.*

6. *si s est dans $\tau_6^{j_6}(\emptyset)$ on a $s \models g$ ou alors $s \models f$ et il existe un état $s' \in S$ tel que $(s,s') \in R$ et $s' \in \tau^{i_6}(\emptyset)$.*

7. *si s est dans $\tau_7^{j_7}(S)$ on a $s \models g$ et alors $s \models f$ ou, pour tout état $s' \in S$ tel que $(s,s') \in R$ on a $s' \in \tau^{i_7}(S)$.*

8. *si s est dans $\tau_8^{j_8}(\emptyset)$ on a $s \models g$ et alors $s \models f$ et il existe un état $s' \in S$ tel que $(s,s') \in R$ et $s' \in \tau^{i_6}(S)$.*

Démonstration. La démonstration de ce lemme est fastidieuse. Nous l'avons faite dans le cas flou et on peut donc trouver sa généralisation au lemme 7. On obtient la preuve du lemme 3 en spécialisant celle du lemme 7, tout simplement.

Dans le prochain lemme, nous montrons que le plus grand point fixe de la fonction $\tau(Z) = f \wedge \mathbf{EX}Z$ est le sous-ensemble de S satisfaisant $\mathbf{EG}f$. On peut cependant le décrire informellement

à partir de la suite

$$S \supseteq \tau(S) \supseteq \tau^2(S) \supseteq \cdots$$

On peut construire $\tau(S)$ à partir de S en ne gardant que les éléments qui satisfont f et le fait qu'ils aient un successeur via R qui satisfait lui-même cette condition, à savoir, satisfaire f et avoir un successeur... Et ainsi de suite. Donc, en définitive, pour tout les éléments satisfaisant l'équation $\tau(Z) = f \wedge \mathbf{EX}Z$, il existe un chemin dans l'arbre des exécutions de K issu de s tel que tous les chemins satisfont f. Réciproquement, si une telle exécution existe à partir d'un élément $s \in S$ alors, clairement, s satisfait la propriété $\tau(Z) = f \wedge \mathbf{EX}Z$. Mais dire qu'il existe un chemin dans l'arbre des exécutions de K issu de s tel que tous les chemins satisfont f c'est dire que s satisfait $\mathbf{EG}f$, tout simplement. Plus formellement, on obtient le lemme suivant.

Lemme 4 *On a:*

1. $\mathbf{AF}f$ *est un point fixe de* $\tau_1(Z) = f \vee \mathbf{AX}Z$
2. $\mathbf{EF}f$ *est un point fixe de* $\tau_2(Z) = f \vee \mathbf{EX}Z$
3. $\mathbf{AG}f$ *est un point fixe de* $\tau_3(Z) = f \wedge \mathbf{AX}Z$
4. $\mathbf{EG}f$ *est un point fixe de* $\tau_4(Z) = f \wedge \mathbf{EX}Z$
5. $\mathbf{A}[f\mathbf{U}g]$ *est un point fixe de* $\tau_5(Z) = g \vee (f \wedge \mathbf{AX}Z)$
6. $\mathbf{E}[f\mathbf{U}g]$ *est un point fixe de* $\tau_6(Z) = g \vee (f \wedge \mathbf{EX}Z)$
7. $\mathbf{A}[f\mathbf{R}g]$ *est un point fixe de* $\tau_7(Z) = g \wedge (f \vee \mathbf{AX}Z)$
8. $\mathbf{E}[f\mathbf{R}g]$ *est un point fixe de* $\tau_8(Z) = g \wedge (f \vee \mathbf{EX}Z)$

Démonstration.

1. Soit $s_0 \models \mathbf{AF}f$. Alors pour toute exécution dans K issue de s_0, il y a un état qui satisfera éventuellement f. Alors soit que f est satisfaite en s_0 soit (non-exclusif) que tous les successeurs de s_0 dans K via R sont des points de départ d'exécutions où f sera éventuellement vraie. Donc on a $s_0 \models f \vee \mathbf{AXAF}f$. Nous venons de montrer que $\mathbf{AF}f \subseteq f \vee \mathbf{AXAF}f$. L'inclusion inverse est évidente: si $s_0 \models f \vee \mathbf{AXAF}f$ alors évidemment $s_0 \models \mathbf{AF}f$, par définition.
2. Soit $s_0 \models \mathbf{EF}f$. Alors il existe une exécution dans K, issue de s_0, dans laquelle il y a un

état qui satisfera éventuellement f. Alors soit que f est satisfaite en s_0 soit (non-exclusif) qu'il existe un successeur de s_0 dans K via R qui est le point de départ d'une exécution où f sera éventuellement vraie. Donc on a $s_0 \models f \vee \mathbf{EXEF}f$. Nous venons de montrer que $\mathbf{EF}f \subseteq f \vee \mathbf{EXEF}f$. L'inclusion inverse est évidente.

3. Soit $s_0 \models \mathbf{AG}f$. Alors pour toute exécution dans K, issue de s_0, tous les états compris dans ces exécutions satisfont f. Ainsi, on a que f est satisfaite non-seulement en s_0 mais aussi en tous les états subséquents de chaque exécution issue d'un successeur de s_0 dans K via R. Donc on a aussi $s_0 \models \mathbf{AXAF}f$. Nous venons de montrer que $\mathbf{AF}f \subseteq f \wedge \mathbf{AXAF}f$. L'inclusion inverse est évidente.

4. Soit $s_0 \models \mathbf{EG}f$. Alors il existe une exécution dans K, issue de s_0, dans laquelle tous les états satisferont f. Ainsi on a que f est satisfaite en s_0 et qu'il existe (au moins) un successeur de s_0 dans K via R qui est le point de départ d'une exécutions où f sera toujours vraie. Donc on a $s_0 \models f \wedge \mathbf{EXEG}f$. Nous venons de montrer que $\mathbf{EG}f \subseteq f \wedge \mathbf{EXEG}f$. L'inclusion inverse est évidente.

5. Soit $s_0 \models \mathbf{A}[f\mathbf{U}g]$. Alors pour toute exécution π dans K, issue de s_0, les états de π satisfont f jusqu'à ce que g le soit (et cela va arriver en un temps fini). Cela entraîne que soit g est satisfaite en s_0 ou (non-exclusif) alors f est satisfaite en s_0 et toutes les exécutions issues des successeurs de s_0 satisfont $\mathbf{A}[f\mathbf{U}g]$. On a donc $s_0 \models g \vee (f \wedge \mathbf{AXA}[f\mathbf{U}g])$ ce qui indique l'inclusion $\mathbf{A}[f\mathbf{U}g] \subseteq g \vee (f \wedge \mathbf{AXA}[f\mathbf{U}g])$. L'inclusion inverse est évidente.

6. Soit $s_0 \models \mathbf{E}[f\mathbf{U}g]$. Alors il existe une exécution π dans K, issue de s_0, dont les états satisfont f jusqu'à ce que g le soit (et cela va arriver en un temps fini). Cela entraîne que soit g est satisfaite en s_0 ou (non-exclusif) alors f est satisfaite en s_0 et il existe une exécution issue d'un successeur de s_0 qui satisfait $\mathbf{E}[f\mathbf{U}g]$. On a donc $s_0 \models g \vee (f \wedge \mathbf{EXE}[f\mathbf{U}g])$ ce qui indique l'inclusion $\mathbf{E}[f\mathbf{U}g] \subseteq g \vee (f \wedge \mathbf{EXE}[f\mathbf{U}g])$. L'inclusion inverse est évidente.

7. Soit $s_0 \models \mathbf{A}[f\mathbf{R}g]$. Alors pour toute exécution π dans K, issue de s_0, on a que les états de π satisfont tous g à partir de s_0 jusqu'au premier état de π où f est *aussi* satisfaite, si cela arrive, car cela n'est pas obligé. Cela entraîne que g est satisfaite en s_0 et soit f l'est aussi, soit tous les successeurs de s_0 se retrouvent dans la même condition que s_0 et donc toute exécution issue d'un successeur de s_0 satisfait $\mathbf{A}[f\mathbf{R}g]$. Ainsi $s_0 \models g \vee (f \wedge \mathbf{AXA}[f\mathbf{R}g])$ ce qui indique l'inclusion $\mathbf{A}[f\mathbf{R}g] \subseteq g \vee (f \wedge \mathbf{AXA}[f\mathbf{R}g])$. L'inclusion inverse est évidente.

8. Soit $s_0 \models \mathbf{E}[f\mathbf{R}g]$. Alors il existe une exécution π dans K, issue de s_0, pour laquelle les états de π satisfont tous g à partir de s_0 jusqu'au premier état de π où f est *aussi* satisfaite, si cela arrive, car cela n'est pas obligé. Cela entraîne que g est satisfaite en s_0 et soit f l'est aussi soit il existe un successeur de s_0 qui se retrouve dans la même condition que s_0 et donc il y a une exécution issue d'un successeur de s_0 satisfaisant $\mathbf{E}[f\mathbf{R}g]$. Ainsi $s_0 \models g \vee (f \wedge \mathbf{EXE}[f\mathbf{R}g])$ ce qui indique l'inclusion $\mathbf{E}[f\mathbf{R}g] \subseteq g \vee (f \wedge \mathbf{EXE}[f\mathbf{R}g])$. L'inclusion inverse est évidente.

Théorème 5 *Soient f, g deux formules de CTL. Alors on a*

1. $\mathbf{AF}f = \mu Z.f \vee \mathbf{AX}Z$
2. $\mathbf{EF}f = \mu Z.f \vee \mathbf{EX}Z$
3. $\mathbf{AG}f = \nu Z.f \wedge \mathbf{AX}Z$
4. $\mathbf{EG}f = \nu Z.f \wedge \mathbf{EX}Z$
5. $\mathbf{A}[f\mathbf{U}g] = \mu Z.g \vee (f \wedge \mathbf{AX}Z)$
6. $\mathbf{E}[f\mathbf{U}g] = \mu Z.g \vee (f \wedge \mathbf{EX}Z)$
7. $\mathbf{A}[f\mathbf{R}g] = \nu Z.g \wedge (f \vee \mathbf{AX}Z)$
8. $\mathbf{E}[f\mathbf{R}g] = \nu Z.g \wedge (f \vee \mathbf{EX}Z)$

Démonstration. Nous avons montré au Lemme 2 que les fonctions τ_i, $1 \leq i \leq 8$ sont monotones. Elles ont toutes, par ailleurs, un domaine fini. Ainsi, par le lemme 1, elles sont $\cup$-*continue* et $\cap$-*continue*. Par le théorème 2 on a alors

$$\mu Z.\tau(Z) = \bigcup_{i \geq 0} \tau^i(\emptyset) \qquad (*)$$

et

$$\nu Z.\tau(Z) = \bigcap_{i \geq 0} \tau^i(S) \qquad (**).$$

Nous allons nous servir de l'identité (*) pour montrer *1* et de l'identité (**) pour montrer *4*. Les preuves pour les cas qui restent sont similaires. On trouvera cependant une preuve complète dans le cas flou aux propositions 9 et 10. Il suffirait alors de les spécialiser pour obtenir une preuve complète du théorème 5.

1. Montrons d'abord que $\bigcup_{i\geq 0} \tau_1^i(\emptyset) \subseteq \mathbf{AF}f$. Cela se fait par récurrence sur i. Clairement on a l'inclusion $\emptyset \subseteq \mathbf{AF}f$. Supposons $\tau_1^i(\emptyset) \subseteq \mathbf{AF}f$. Puisque τ_1 est monotone et que $\mathbf{AF}f$ est un point fixe de τ_1 on a

$$\tau_1^{i+1}(\emptyset) \subseteq \tau_1(\mathbf{AF}f) = \mathbf{AF}f \Rightarrow$$

$$\tau_1^{i+1}(\emptyset) \subseteq \mathbf{AF}f \Rightarrow \forall i \geq 0,\ \tau^{i+1}(\emptyset) \subseteq \mathbf{AF}f \Rightarrow \bigcup_{i\geq 0} \tau_1^i(\emptyset) \subseteq \mathbf{AF}f.$$

Montrons maintenant $\mathbf{AF}f \subseteq \bigcup_{i\geq 0} \tau_1^i(\emptyset)$. Soit $s \in \mathbf{AF}f$. On veut montrer qu'il existe un entier $i \geq 0$ tel que $s \in \tau_1^i(\emptyset)$.

Examinons d'abord la nature de $\tau_1^i(\emptyset)$. Pour $i = 1$, $\tau_1^1(\emptyset)$ est constitué de tous les états $s \in S$ satisfaisant la propriété f, de même que tous les états $x \in S$ à partir desquels toutes les exécutions satisferont f (en un temps fini). Pour $i = 2$, $\tau_1^2(\emptyset)$ contient $\tau_1(\emptyset)$, tous les états $x \in S$ pour lesquels il existe $s \in S$ tel que $f \in L(s)$ et $(x, s) \in R$ et, de même, encore ici, que tous les états $x \in S$ à partir desquels toutes les exécutions satisferont f (en un temps fini) via R. De proche en proche, on constate que pour tout entier $i \geq 1$

$$\tau_1^i(\emptyset) = \{s \in S | \forall \pi\ t.q\ \pi[1] = s\ \exists j,\ 0 \leq j < i, \pi[j] \models f\} \cup\ X$$

où X est l'ensemble de tous les états $x \in S$ à partir desquels toutes les exécutions satisferont f (en un temps fini).

Maintenant, soit $s \in \mathbf{AF}f$. On sait que cette appartenance signifie que toutes les exécutions issues de s dans K satisferont f éventuellement. Prenons le plus long préfixe issu de s satisfaisant f pour une première fois. Puisque toutes les exécutions issues de s satisferont éventuellement f, ce préfixe est bien défini. S'il est de longeur i, il est clair que $s \in \tau_1^{i+1}(\emptyset)$.

2. Dans un sens, cela se fait par induction sur i. En effet on a bien $\mathbf{EG}f \subseteq S$. Supposons que l'on ait $\mathbf{EG}f \subseteq \tau^n(S)$ vraie pour un certain entier $n \geq 0$. Alors puisque τ est monotone, on a $\tau(\mathbf{EG}f) \subseteq \tau^{n+1}(S)$. Puisque $\mathbf{EG}f$ est un point fixe de τ on a $\tau(\mathbf{EG}f) = \mathbf{EG}f$ et donc $\mathbf{EG}f \subseteq \tau^{n+1}(S)$. Ainsi, pour tout entier $n \geq 0$, on a $\mathbf{EG}f \subseteq \tau^n(S)$ et finalement $\mathbf{EG}f \subseteq \cap_n \tau^n(S)$.

Dans l'autre sens, on veut montrer que $\cap_n \tau^n(S) \subseteq \mathbf{EG}f$. Pour cela supposons $s \in \cap_n \tau^n(S)$. Alors, pour tout entier $i \geq 0$ on a $s \in \tau^i(S)$ ce qui entraîne $s \in \tau^{i_0}(S)$. Par le Lemme 4 s est le point de départ d'une exécution dans K où chaque état satisfait f. Donc on a $s \models \mathbf{EG}f$.

CHAPITRE III

LES LOGIQUES ET SYSTÈMES FLOUS

D'un point de vue mathématique, la logique floue permet d'envisager le passage, en logique, du cas discret au cas infini. Notre but est de généraliser la vérification de modèles à ce niveau. Dans ce chapitre nous présentons d'abord un bref aperçu de certains concepts de base de la logique floue. C'est l'objet de la section 3.1. À partir de là, il est possible de définir une structure de Kripke floue, ce que nous faisons dans la section 3.2, sans laquelle la représentation floue de modèles serait impossible.

Nous poursuivons en présentant en 3.3 le coeur de notre mémoire, les logiques $\aleph CTL^*$ et $\aleph CTL$ qui constituent l'analogue flou standard des logiques CTL^* et CTL ordinaires. La dernière section, quant à elle, est consacrée à l'existence des points fixes flous, condition requise pour pouvoir effectuer de la vérification floue de modèles.

3.1 La logique floue

Définition 24 Soit E un ensemble de référence. Un *ensemble flou* sur E est une fonction

$$\delta_E : E \to [0,1].$$

La collection de tous les ensembles flous sur E est dénoté $\aleph(E)$. Il s'agit de l'ensemble de toutes les fonctions de E dans $[0,1]$,

$$\aleph(E) = [0,1]^E.$$

On peut considérer les ensembles flous comme des généralisations des fonctions caractéristiques χ d'ensembles usuels. En effet, prenons E un ensemble de référence et $U \subseteq E$ un sous-ensemble

arbitraire de E. Alors la fonction caractéristique de U dans E est donnée par la règle suivante:

$$\chi_E^U : E \to \{0,1\}$$
$$\forall x \in E,\ \chi_E^U(x) = \begin{cases} 0 & \text{si } x \notin U, \\ 1 & \text{sinon.} \end{cases}$$

Le fait d'ajouter tous les réels entre 0 et 1 au codomaine de la fonction caractéristique ordinaire permet d'en enrichir le contexte en interprétant l'image de cette fonction comme un *degré d'appartenance* (δ) d'un élément à un ensemble plutôt que de simplement «être ou ne pas être» élément d'un ensemble, comme c'est le cas en théorie classique des ensembles.

D'un point de vue logique, un ensemble flou constitue aussi une généralisation du prédicat binaire

$$\chi_E^U : E \to \mathbf{B}$$
$$\forall x \in E,\ \chi_E^U(x) = \begin{cases} \textbf{faux} & \text{si } x \notin U, \\ \textbf{vrai} & \text{sinon.} \end{cases}$$

Dans ce contexte, la logique floue devient une logique multivalente à valeurs dans $[0,1]$ où l'image de $x \in E$ s'interprète alors comme le *degré de vérité* de x. Il faut bien entendu ajuster et généraliser les connecteurs et quantificateurs logiques usuels pour qu'ils aient un sens dans cette théorie. Il y a plusieurs manières de faire.

Avant de passer aux définitions des opérations de la logique floue, on doit ajouter qu'il est d'usage lorsqu'on veut généraliser une théorie de faire en sorte que l'on obtienne une théorie «augmentée» de la première, dans le sens où on essaie de préserver l'ensemble des propriétés originales de la théorie qui existe déjà. Ces dernières propositions devraient correspondre à des spécialisations dans la théorie généralisée. Il faut donc trouver un bon contexte où les généralisations sont possibles.

Par exemple, en logique propositionnelle, la conjonction et la disjonction des prédicats sont le plus souvent définies via leur table de vérité. De même, en théorie des ensembles, on définit à l'aide de règles précises l'intersection et la réunion des ensembles qui sont les opérations associées respectivement à la conjonction et à la disjonction logiques. Mais ces définitions sont difficilement généralisables au contexte flou directement. Les fonctions de vérité et les fonctions caractéristiques semblent plus appropriées. Ainsi, les conjonction et disjonction, ou leurs analogues ensemblistes, peuvent être explicitées de la manière suivante. Disons d'abord qu'au lieu de prendre **vrai** et **faux** comme valeur de vérité, on prend plutôt les valeurs numériques respectives 1 et 0. Ensuite, soient $\chi_1 : E \to \{0,\ 1\}$ et $\chi_2 : E \to \{0,\ 1\}$ deux prédicats. On

définit le «et» logique

$$\chi_1 \wedge \chi_2 : E \rightarrow \{0,\ 1\}$$

par

$$\forall x \in E,\ \chi_1 \wedge \chi_2(x) = \mathrm{Min}(\chi_1(x), \chi_2(x)).$$

De même on définit le «ou» logique

$$\chi_1 \vee \chi_2 : E \rightarrow \{0,\ 1\}$$

par

$$\forall x \in E,\ \chi_1 \vee \chi_2(x) = \mathrm{Max}(\chi_1(x), \chi_2(x)).$$

Dans la définition qui suit, les opérateurs de conjonction et disjonction floues sont calqués sur les définitions qui précèdent. Il est à noter que ces opérations sont définies ponctuellement, de sorte que l'opération est bien définie, les minimum et maximum de deux nombres réels existant toujours.

Définition 25 Soient E un ensemble de référence, $\delta, \delta_1, \delta_2 : E \rightarrow [0,1]$ trois ensembles flous. Nous définissons les connecteurs logiques flous suivants:

1. $\wedge$: le «et» (conjonction) de notre logique floue est défini comme étant la fonction minimum entre ses deux arguments:

$$\delta_1 \wedge \delta_2 : E \rightarrow [0,\ 1],$$

$$\forall x \in E : \delta_1 \wedge \delta_2(x) = \mathrm{Min}(\delta_1(x), \delta_2(x)).$$

2. $\vee$: le «ou» (disjonction) de notre logique floue est défini comme étant la fonction maximum entre ses deux arguments:

$$\delta_1 \vee \delta_2 : E \rightarrow [0,\ 1],$$

$$\forall x \in E : \delta_1 \vee \delta_2(x) = \mathrm{Max}(\delta_1(x), \delta_2(x)).$$

3. $\sim$: le «non» (négation) de notre logique floue est défini comme étant la fonction complémentaire relativement à 1 :

$$\sim \delta : E \rightarrow [0,\ 1],$$

$$\forall x \in E : \sim \delta(x) = 1 - (\delta(x)).$$

La généralisation précédente, dont on verra dans ce mémoire les effets sur la vérification de modèles, n'est pas la seule possible en logique floue, loin de là. La notion centrale permettant de donner plusieurs sens à la conjonction et aux autres connecteurs logiques en général est celle de *t-norme* (voir, par exemple, (8), p.28). Son étude dépasse largement le cadre de notre travail. Nous ne l'aborderons pas et nous nous en tiendrons à la logique floue dite «standard» ou «usuelle» décrite par la définition 25.

En logique du premier ordre, on considère aussi les quantificateurs *existentiel*, $\exists$, et *universel*, $\forall$. Lorsqu'on veut quantifier un prédicat Π défini sur un ensemble E, on examine sa véracité ponctuellement sur les éléments de E. On a

- $(\exists x \in E, \Pi(x))$ est vraie si, et seulement si, $\Pi(x)$ est vraie pour au moins un x dans E,
- $(\forall x \in E, \Pi(x))$ est vraie si, et seulement si, $\Pi(x)$ est vraie pour tout x dans E.

Ces quantificateurs peuvent aussi être «numérisés» en considérant $\Pi : E \to \{0,\ 1\}$ comme une fonction à valeur dans $\{0,\ 1\}$. On obtient alors

$$(\exists x \in E, \Pi(x)) = 1 \Leftrightarrow \max_{x \in E} \Pi(x) = 1,$$

$$(\forall x \in E, \Pi(x)) = 1 \Leftrightarrow \min_{x \in E} \Pi(x) = 1.$$

Ces dernières formules se généralisent aisément au cas flou standard en considérant maintenant $\Pi : E \to [0,\ 1]$ comme une fonction à valeurs dans l'intervalle $[0,\ 1]$. On obtient la définition suivante.

Définition 26 Soit E un ensemble et $c, c' \in [0,\ 1]$. Alors on définit le quantificateur existentiel flou (standard) par

$$(\exists x \in E, \Pi(x)) = c \Leftrightarrow \max_{x \in E} \Pi(x) = c$$

et le quantificateur universel flou (standard) par

$$(\forall x \in E, \Pi(x)) = c' \Leftrightarrow \min_{x \in E} \Pi(x) = c'.$$

Définition 27 Soient A et B deux ensembles (classiques). Une *relation floue* R sur A et B est un sous-ensemble flou du produit cartésien $A \times B$ de A et B:

$$R : A \times B \to [0, 1].$$

3.2 Les structures de Kripke floues

Définition 28 *Une structure de Kripke floue* $K_{\aleph}$ sur un ensemble de propositions atomiques PA est un quadruplet $K_{\aleph} = (S, S_0, R_{\aleph}, L_{\aleph})$ où

i. S est un ensemble (ordinaire) *fini* d'états,

ii. $S_0 \subseteq S$ est l'ensemble des états initiaux de $K_{\aleph}$,

iii. $R_{\aleph} \subseteq S \times S$ est une relation floue binaire totale sur S, dite *relation de transition* de $K_{\aleph}$. Par *totale*, on veut dire que

$$\forall s \in S,\ \exists s' \in S,\ R_{\aleph}(s,\ s') \neq 0.$$

iv. $L_{\aleph} : S \rightarrow [0,1]^{PA}$ est un fonction qui assigne à chaque état de $s \in S$ une fonction de vérité floue $I_s : PA \rightarrow [0,1]$ assignant à chaque élément de PA une valeur de vérité comprise entre 0 et 1, inclusivement.

Si $a \in PA$ est une proposition atomique alors la fonction $L_{\aleph}(s)(a)$ est un nombre réel compris entre 0 et 1, indiquant le degré de vérité (d'appartenance) de a dans l'état s.

Définition 29 Soit $K_{\aleph}$ une structure de Kripke floue sur PA. Une *exécution floue* dans $K_{\aleph}$ est une suite infinie $\pi = s_0 s_1 s_2 s_3 \ldots$ d'états de S où pour tout entier $i \geq 0$, on a $R_{\aleph}(s_i, s_{i+1}) > 0$. On appelle aussi cette exécution un *chemin* dans $K_{\aleph}$. L'ensemble des chemins que l'on peut engendrer avec $K_{\aleph}$ est dénoté $\mu(K_{\aleph})$.

3.3 Les logiques temporelles floues

3.3.1 La logique floue $\aleph CTL^*$

La généralisation de la logique CTL^* au contexte flou que nous dénotons $\aleph CTL^*$ et que nous proposons ici est en partie similaire à la généralisation de CTL au contexte multivarié que l'on peut trouver dans les travaux de Chechik (2). Cependant, puisque les opérateurs temporels de CTL^* peuvent être utilisés indépendamment les uns des autres, contrairement à ceux de CTL qui viennent toujours par paire, nous devons tous les interpréter un à un.

Ce travail n'est pas inutile. Une fois la logique $\aleph CTL^*$ définie, il suffit en effet de *composer* systématiquement certains de ces opérateurs pour obtenir $\aleph CTL$ entièrement.

Pour effectuer la généralisation de CTL^*, nous pourrions d'abord nous donner une structure de Kripke floue, puis définir une relation de satisfaction "$\models_x$", où x est un nombre réel donné tel que $0 \leq x \leq 1$. Avec cette structure de Kripke floue $K_\aleph$ donnée, $s \in S_\aleph$ un état de l'ensemble des états de $K_\aleph$ et f une formule d'état ou de chemin, la signification de

$$K_\aleph, s \models_x f$$

serait alors que la structure de Kripke floue $K_\aleph$ satisfait la formule f en un état s à un certain degré $x \in [0,1]$, degré étant entendu au sens flou du terme.

Il est plus facile cependant de procéder avec des fonctions de vérité, pour des raisons déjà mentionnées. Ainsi, plutôt que procéder avec la relation "$\models_x$", nous définissons une fonction d'évaluation qui généralise celle de la définition 23 de la manière suivante (nous la dénotons aussi e, par abus) :

$$\mathbf{e} : S_\aleph \times \Phi(AP) \to [0,1], \quad (s,f) \mapsto \mathbf{e}(s,f) = \mathbf{e}_s(f)$$

et, avec abus,

$$\mathbf{e} : \mu(K_\aleph) \times \Upsilon(AP) \to [0,1], \quad (\pi,g) \mapsto \mathbf{e}(\pi,g) = \mathbf{e}_\pi(g).$$

Avant de donner les valeurs de vérité des différentes formules, nous justifions les choix que nous avons faits pour arriver à cette forme floue. Il faut noter que la généralisation que nous allons décrire est issue du problème purement théorique de généraliser la vérification de modèles ordinaire à la logique floue, et non pas d'un problème concret particulier. Dans ce contexte libre de toute représentation «réelle», la tâche consiste à relever systématiquement à la logique floue les divers aspects de la vérification de ces modèles usuels. Comme nous avons déjà introduit la notion de structure de Kripke floue, il reste à savoir comment réagit cette nouvelle définition relativement aux opérations logiques usuelles dans la logique CTL^*.

Remarque 3 Avant d'introduire la sémantique des opérateurs et des formules, il faudrait normalement en préciser la syntaxe. Mais comme elle est exactement la même que celle donnée dans les définitions 20 et 21, nous ne la réécrirons pas inutilement. Les formules seront interprétées plus loin sur un ensemble d'états ou sur un ensemble de chemins d'une structure de Kripke floue. Ces deux derniers étant des ensembles au sens classique du terme, il n'y a pas lieu de redéfinir quoi que ce soit.

Plusieurs opérations demeurent sensiblement les mêmes, particulièrement au niveau de leur écriture. Celles correspondant aux opérations de la logique de chemin se généralisent bien au cas flou, sans qu'il y ait même à changer quoi que ce soit dans la définition 23. Cette définition ne demande qu'un seul ajustement léger où on tient compte explicitement du *degré d'appartenance* des éléments aux ensembles. En effet, le point 1, fondamental à cet égard,

$$\forall p \in PA, \quad \mathbf{e}_s(p) = 1 \quad \underset{\text{def}}{\Leftrightarrow} \quad p \in L(s)$$

s'écrira plutôt

$$\forall p \in PA, \quad \mathbf{e}_s(p) = x \quad \underset{\text{def}}{\Leftrightarrow} \quad L(s)(p) = x$$

sans qu'il soit vraiment nécessaire de justifier davantage.

Les points 2 à 15, inclus, s'écrivent exactement de la même manière dans le cas flou. Prendre les fonctions Min et Max pour définir les opérations logiques suffit à régler les difficultés propres à notre généralisation. Cela est vrai, entre autres, pour les quantificateurs existentiel et universel **E** et **A**. On doit noter que notre travail se démarque ici complètement de la généralisation de Chechik au cas multivarié. Nous faisons le choix de ne pas tenir compte de la relation de transition $R_{\aleph}$ pour comptabiliser la valeur de vérité des opérateurs temporels. Tout au plus demandons nous, comme dans le cas ordinaire, qu'il y ait un lien *non-nul* entre les états du système pour passer d'un état à un autre. Ainsi, à l'instar du traitement ordinaire, lorsque nous supposons l'existence d'un chemin π, nous supposons que le lien entre chacun des états successifs du chemin π est une valeur *strictement* plus grande que 0. En symboles mathématiques on pourrait écrire

$$\exists \pi \Leftrightarrow \forall i \geq 0, R_{\aleph}(\pi[i], \pi[i+1]) > 0.$$

Nous croyons que cette généralisation est viable car rien, *a priori*, ne lie l'aspect flou de la relation $R_{\aleph}$ à celui de la sémantique des propositions atomiques.

Ce choix n'est pas sans conséquence sur la pertinence de la définition 23. En effet, nous pouvons maitenant intégrer au cas flou les définitions qu'on y trouve sans en changer le moindrement l'écriture.

Bien entendu, la sémantique des opérateurs $\aleph CTL^*$ et CTL^* peut différer du tout au tout, puisque les valeurs possibles prises par les premiers constituent un segment complet alors qu'il n'y en a que deux permises pour le second opérateur. Il est donc souhaitable de les passer en revue pour en préciser le sens. Notons aussi que la logique $\aleph CTL^*$ généralise correctement celle de CTL^* car en spécialisant la première on obtient la seconde.

La sémantique de $\aleph CTL^*$

Dans la liste suivante, on se réfère à la définition 22 pour les points de 1 à 15.

- **Le point 1.** Le point 1 a été discuté plus haut. Il appert que c'est le seul point dans les 15 donnés dont l'écriture ait besoin d'être modifiée.

- **Le points 2–10.** Au sujet des points 2 à 10 il y a peu à dire si ce n'est que l'on a affaire à des opérateurs flous ordinaires, qu'ils soient comptabilisés sur des états ou des exécutions. On peut remarquer aussi au point 7 que la fonction caractéristique χ est à valeur dans $\{0,1\}$ et non pas dans $[0,1]$. Cela signifie qu'un état appartient ou n'appartient pas à un chemin π. Il n'y a pas de degré d'appartenance dans le cas présent.

- **Le point 11.** La formule

$$\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{X}g) = \mathbf{e}_{\pi^1}(g)$$

signifie que la fonction de vérité de la formule $(\mathbf{X}g)$ en $\pi[0]$ est calculée à partir de la fonction de vérité de g en $\pi[1]$, comme c'est le cas en logique CTL^* ordinaire, avec la différence que cette fonction prend ses valeurs dans $[0,\ 1]$ plutôt que dans $\{0,\ 1\}$. Insistons sur le fait qu'il suffit que le lien $R_\aleph(\pi[0], \pi[1])$ soit non-nul, *aussi ténu soit-il*, pour que la valeur de vérité de cette formule soit donnée *uniquement* par la fonction de vérité de g en $\pi[1]$.

- **Le point 12.** La formule

$$\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{F}g) = \max_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

est aussi reprise telle quelle, sans modification dans l'écriture. Cependant la sémantique de l'opération est ici une vraie généralisation et demande à être précisée. En effet cet opérateur, dans le cas ordinaire, retourne **1** (vrai) si jamais la fonction de vérité de g donne 1 (vrai) en un des états de l'exécution π. Dans le cas flou que nous proposons ici, il retourne le *maximum* des valeurs de vérités atteintes par la fonction de vérité de g sur les divers états de l'exécution π, qui n'est pas forcément **1**.

- **Le point 13.** La formule

$$\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{G}g) = \min_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

est reprise telle quelle, sans modification dans l'écriture. Les mêmes remarques qu'au point 12 s'appliquent ici, en changeant le terme *maximum* qui s'y trouve par le terme *minimum*.

Dans le cas ordinaire, l'opérateur temporel $(g_1 \mathbf{U} g_2)$ tient compte de l'évaluation des valeurs de vérités des propriétés g_1 et g_2 du *début* $\pi[0]$ d'un chemin π jusqu'à un *certain* état $\pi[j]$. Toutes les évaluations de g_1 doivent êtres égales à 1 (vraies) à partir de $\pi[0]$ jusqu'à ce que celle de g_2 le soit également.

Il y a plusieurs généralisations possibles de cet opérateur au cas flou. Nous optons pour celle consistant à demander d'abord à ce que les valeurs de vérité de la propriété g_1 soient *non-fausses* plutôt que vraies. Ainsi, pour tout entier $0 \leq k < j$, on utilisera l'inégalité $\mathbf{e}_{\pi[k]}(g_1) > 0$. Cette condition est naturelle dans le cas flou et correspond d'une certaine manière à choisir des transitions non-nulles pour passer d'un état à un autre dans la relation floue.

- **Le point 14.** La formule

$$\max_{k \in \mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \leq j < k}\ \mathbf{e}_{\pi^j}(g_1)))$$

est reprise telle quelle, sans modification dans l'écriture. Le sens de cet opérateur est le suivant. L'évaluation $\mathbf{e}_\pi(g_1 \mathbf{U} g_2)$ est supérieure à 0 si et seulement si il existe un entier $k \geq 0$ tel que l'évaluation $\mathbf{e}_{\pi^k}(g_2)$ est supérieure à 0 et tel que quelque soit l'entier naturel j strictement inférieur à k, l'évaluation $\mathbf{e}_{\pi^j}(g_1)$ est supérieure à 0.

Il est important de noter que l'évaluation de $\mathbf{e}_\pi(g_1 \mathbf{U} g_2)$ retourne le maximum des valeurs minimales calculées parmi l'ensemble des suites d'états de la forme

$$(\mathbf{e}_{\pi^0}(g_1), \mathbf{e}_{\pi^1}(g_1), \ldots, \mathbf{e}_{\pi^{k-1}}(g_1), \mathbf{e}_{\pi^k}(g_2)).$$

Ce maximum n'est pas nécessairement localisé au premier index j de la suite où l'évaluation $\mathbf{e}_{\pi^j}(g_2)$ est plus grande que 0. En fait, il n'est même pas nécessaire qu'il retourne une valeur calculée à partir de g_2. Le minimum peut très bien n'être atteint que par l'évaluation en un état donné de g_1.

- **Le point 15.** La formule

$$\min_{k \in \mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g_2), \max_{0 \leq j < k}\ \mathbf{e}_{\pi^j}(g_1)))$$

est reprise telle quelle, sans modification dans l'écriture. Cet opérateur a été défini pour remplir le rôle d'opérateur dual à **U**.

Définition 30 Soient $K_{\aleph} =< S,\ S_0,\ R_{\aleph},\ L_{\aleph} >$ une structure de Kripke *floue* sur un ensemble PA de proposition atomiques, s un état de S. Soit $x \in [0,1]$ un nombre réel compris inclusivement entre 0 et 1. La fonction de vérité floue sur K de $\aleph CTL^*$ est définie ainsi:

1. $\forall p \in PA, \quad \mathbf{e}_s(p) = x \underset{\text{def}}{\Leftrightarrow} L(s)(p) = x.$

2. $\forall f \in \Phi(PA), \quad \mathbf{e}_s(\sim f) = 1 - \mathbf{e}_s(f).$

3. $\forall f_1, f_2 \in \Phi(PA), \quad \mathbf{e}_s(f_1 \vee f_2) = \mathrm{Max}(\mathbf{e}_s(f_1), \mathbf{e}_s(f_2)).$

4. $\forall f_1, f_2 \in \Phi(PA), \quad \mathbf{e}_s(f_1 \wedge f_2) = \mathrm{Min}(\mathbf{e}_s(f_1), \mathbf{e}_s(f_2)).$

5. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_s(\mathbf{E}g) = \max_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(g).$

6. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_s(\mathbf{A}g) = \min_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(g).$

7. $\forall f \in \Phi(PA), \quad \mathbf{e}_\pi(f) = \mathrm{Min}(\chi(\pi[0] = s), \mathbf{e}_s(f)).$

8. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\sim f) = 1 - \mathbf{e}_\pi(g).$

9. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \wedge g_2) = \mathrm{Min}(\mathbf{e}_\pi(g_1), \mathbf{e}_\pi(g_2)).$

10. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \vee g_2) = \mathrm{Max}(\mathbf{e}_\pi(g_1), \mathbf{e}_\pi(g_2)).$

11. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{X}g) = \mathbf{e}_{\pi^1}(g).$

12. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{F}g) = \max_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g).$

13. $\forall g \in \Upsilon(PA), \quad \mathbf{e}_\pi(\mathbf{G}g) = \min_{j \in \mathbf{N}} \mathbf{e}_{\pi^j}(g).$

14. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \mathbf{U} g_2) = \max_{k \in \mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g_2), \min_{0 \le j < k} \mathbf{e}_{\pi^j}(g_1))).$

15. $\forall g_1, g_2 \in \Upsilon(PA), \quad \mathbf{e}_\pi(g_1 \mathbf{R} g_2) = \min_{k \in \mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g_2), \max_{0 \le j < k} \mathbf{e}_{\pi^j}(g_1))).$

Remarque 4 Dans (15) on définit l'évaluation l'opérateur **EX** en un seul «bloc» par

$$\mathbf{e}_s(\mathbf{EX}f) = \max_{\pi\in\mu(K),\pi[0]=s} (\min_{t\in S}(R(s,t), \mathbf{e}_t(f)).$$

Si, dans $\aleph CTL^*$ on définit l'opérateur $\mathbf{X_R}$ par

$$\mathbf{e}_\pi(\mathbf{X_R}f) = \min_{\pi\in\mu(K)} (\mathbf{e}_{\pi[1]}(f), R(\pi[0],\pi[1])),$$

alors on obtient

$$\mathbf{EX} = \mathbf{EX_R}.$$

3.3.2 La logique floue $\aleph CTL$

Dans (3) et (2), on définit la logique temporelle CTL indépendamment (formellement du moins) de CTL^*. Dans le chapitre 2, section 4.3 de ce mémoire, nous avons plutôt adopté le point de vue inverse en définissant CTL comme une spécialisation de CTL^*, à l'instar de ce que l'on peut trouver dans (1, p. 64). C'est cette dernière voie que nous adoptons pour définir notre généralisation floue $\aleph CTL$ de la logique temporelle CTL.

Rappelons que la logique CTL exprime uniquement des propriétés d'états d'une structure de Kripke ordinaire. Il en sera de même de $\aleph CTL$, pour une structure de Kripke floue. Cependant, plutôt que définir les opérateurs $\aleph CTL$ en blocs insécables composés toujours d'un quantificateur (opérateur) logique **E** ou **A** suivi d'un des opérateurs temporels **X, F, G, R, U,** comme cela est fait dans (3) et (2), nous procédons plutôt en «composant» deux opérateurs de la logique CTL^*. Pour ce faire, soulignons d'abord que toute formule d'état est, *par définition*, une formule de chemin. Ainsi, appliquer un des opérateurs **X, F, G** à une formule d'état (f) arbitraire ou un des opérateurs **U, R** à une paire (f,g) arbitraire de formules a du sens, syntaxiquement à tout le moins. Puisque dans chaque cas le résultat de cette application est une formule de chemin, il est permis d'écrire, quelques soient les formules d'états f,g sur $K_\aleph$

$$(Op_1 Op_2)(f) = Op_1(Op_2(f)),$$

où $Op_1 \in \{\mathbf{A},\ \mathbf{E}\}$ et $Op_2 \in \{\mathbf{X},\ \mathbf{F},\ \mathbf{G}\}$ et

$$(Op_1 Op_2)(f,g) = Op_1(Op_2(f,g)),$$

où $Op_1 \in \{\mathbf{A},\ \mathbf{E}\}$ et $Op_2 \in \{\mathbf{U},\mathbf{R}\}$. Par exemple, on définit $\mathbf{AX}$ par la formule

$$\forall f \in \Phi(PA), \quad \mathbf{AX}(f) = \mathbf{A}(\mathbf{X}(f)).$$

Le calcul des fonctions de vérité (la sémantique) de $\aleph CTL$ se fait donc à partir de celle, plus générale, de $\aleph CTL^*$.

3.4 Les points fixes flous

Dans cette section nous adaptons la théorie des points fixes au cas flou, permettant ainsi de faire une généralisation de l'approche itérative de la vérification de modèles. Cette généralisation repose sur la théorie classique des ensembles que l'on appellera des ensembles *rigides*. Les opérateurs de points fixes que nous verrons auront comme arguments des ensembles rigides et donc comme image des ensembles rigides aussi. Cependant, les formules données pour définir ces opérateurs seront floues en général. Pour pouvoir traiter cette «mixture» convenablement, il ne faut pas oublier qu'un ensemble rigide est tout simplement un cas spécial d'ensemble flou, ne prenant comme image que les valeurs 0 ou 1 du codomaine $[0,\ 1]$. Il n'est pas difficile dans ce contexte de faire des liens avec des ensembles rigides.

Soit S un ensemble fini d'états d'une structure de Kripke floue, $Z \subseteq S$ un sous-ensemble de S. Alors la notion de sous-ensemble peut être vue comme une propriété, une formule (au sens strict) d'états du système. La valeur de vérité de Z *dans* S est définie par

$$\mathbf{e}_s(Z) = 1 \Leftrightarrow s \in Z.$$

De plus on peut généraliser l'identification des formules f de CTL aux prédicats ensemblistes

$$f \equiv \{s \in S | K, s \models f\}$$

à $\aleph CTL$ en posant

$$f \equiv \{s \in S | \mathbf{e}_s(f) > 0\}.$$

Il est fort probable que la définition précédente soit restrictive. Une autre généralisation, beaucoup plus vaste, pourrait être étudiée. En indiçant par un nombre x entre 0 et 1 la relation précédente on obtient

$$f \equiv_x \{s \in S | \mathbf{e}_s(f) > x\}.$$

Avec les définitions précédentes, les huit opérateurs de prédicats de la définition 23 peuvent être revus dans un contexte flou. Ce sont toujours des transformations de prédicats ensemblistes rigides dans le sens où pour i, on a $\tau_i : \wp(S) \to \wp(S)$.

Pour distinguer le contexte flou du contexte ordinaire on pose, pour tout entier i, que l'analogue flou de τ_i s'écrira $\tau_i^{\aleph}$. On a donc

1. $\tau_1^{\aleph}(Z) = f \vee \mathbf{AX}Z$

2. $\tau_2^{\aleph}(Z) = f \vee \mathbf{EX}Z$

3. $\tau_3^{\aleph}(Z) = f \wedge \mathbf{AX}Z$

4. $\tau_4^{\aleph}(Z) = f \wedge \mathbf{EX}Z$

5. $\tau_5^{\aleph}(Z) = g \vee (f \wedge \mathbf{AX}Z)$

6. $\tau_6^{\aleph}(Z) = g \vee (f \wedge \mathbf{EX}Z)$

7. $\tau_7^{\aleph}(Z) = g \wedge (f \vee \mathbf{AX}Z)$

8. $\tau_8^{\aleph}(Z) = g \wedge (f \vee \mathbf{EX}Z)$

Ces opérateurs se *calculent*, par exemple dans le cas de $\tau_1^{\aleph}$, de la manière suivante:

1. Tout élément $s \in S$ est dans $\tau_1^{\aleph}(Z)$ si et seulement si

- $\mathbf{e}_s(f) > 0$,

ou

- $\mathbf{e}_s(\mathbf{AX}Z) > 0$.

où la dernière condition se calcule comme suit:

$$\mathbf{e}_s(\mathbf{AX}Z) = \min_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_\pi(\mathbf{X}Z) = \min_{\pi \in \mu(K), \pi[0]=s} \mathbf{e}_{\pi[1]}(Z) > 0.$$

L'interprétation de cette formule est claire dans la première clause de la disjonction. Pour la seconde, on a que l'état s satisfait cette condition si pour tout chemin issu de s le second état de ces chemins est dans Z.

Comme on peut le constater, l'interprétation de la formule est la même que dans le cas rigide. Il en va de même pour les transformations de prédicats $\tau_i^{\aleph}, 2 \leq i \leq 8$.

Nous allons maintenant examiner les itérées successives de ces transformations. Nous allons grandement profiter de cette étude car nous allons montrer ensuite que toutes les transformations $\tau_i^{\aleph}, 1 \leq i \leq 8$, sont monotones. Ainsi leurs plus petits et plus grands points fixes pourront être calculés récursivement comme des limites *finies* de réunions ou intersections d'itérées.

Dans ce qui suit, le cheminement est le même que dans le cas rigide: dans les lemmes 5 et 6 on décrit la nature des itérées des transformations $\tau_i^{\aleph}$ — analogue de la liste suivant la définition 23 —, à la proposition 7 on démontre la monotonicité des transformationn — analogue du lemme

2—, au lemme 7 on décrit la nature des points fixes des transformations - analogue du lemme 3 —, à la proposition 8 on démontre que les point fixes des transformations ne sont nuls autres que les opérateurs $\aleph CTL$ souhaités - analogue du lemme 4 —, et finalement, aux propositions 9 et 10, on démontre que ces points fixes sont respectivement les plus petits et plus grands points fixes des transformations $\tau_i^{\aleph}$ — analogue du théorème 5 — .

Lemme 5 (Propriétés des itérées sur $\emptyset$) *Soit S l'ensemble de tous les états d'un système et $i \in \{1,\ 2,\ 5,\ 6\}$ alors la j-ème itérée des transformations de prédicats $(\tau_i^{\aleph})^j$ peut être décrite ainsi:*

1. $(\tau_1^{\aleph})^j(\emptyset)$ *est le sous-ensemble des éléments x de S tels que pour toute exécution π de longueur j issue de x il y ait au moins un des états $\pi[k], 0 \leq k \leq j$, de l'exécution qui satisfasse* $\mathbf{e}_{\pi[k]}(f) > 0$.

2. $(\tau_2^{\aleph})^j(\emptyset)$ *est le sous-ensemble des éléments x de S pour lesquels il existe une exécution π de longueur j issue de x pour laquelle y a au moins un des états $\pi[k], 0 \leq k \leq j$, de l'exécution qui satisfait* $\mathbf{e}_{\pi[k]}(f) > 0$.

3. $(\tau_5^{\aleph})^j(\emptyset)$ *est le sous-ensemble des éléments x de S tels que tout chemin π issu de x de longueur j doit satisfaire la condition* $\mathbf{e}_{\pi[k]}(f) > 0, 0 \leq k \leq j$, *jusqu'à ce que la condition* $\mathbf{e}_{\pi[m]}(g) > 0, 0 \leq m \leq j$, *soit vraie.*

4. $(\tau_6^{\aleph})^j(\emptyset)$ *est le sous-ensemble des éléments x de S tels qu'il existe un chemin π issu de x de longueur j qui doit satisfaire la condition* $\mathbf{e}_{\pi[k]}(f) > 0, 0 \leq k \leq j$, *jusqu'à ce que la condition* $\mathbf{e}_{\pi[m]}(g) > 0, 0 \leq m \leq j$, *soit vraie.*

Remarque 5 Nous ne démontrons que la première propriété du Lemme 5, à titre indicatif. Toutes les autres se font essentiellement de la même façon. On voit bien apparaître la nature des points fixes, et ce dès les premières itérations.

Démonstration de 1. Procédons par récurrence sur l'index j des itérées de la transformation. Soit $j \geq 0$ un entier et $s \in S$. Considérons les itérées $(\tau_1^{\aleph})^j(\emptyset)$.

Pour $j = 0$ on a $(\tau_1^{\aleph})^0(\emptyset) = \emptyset$. L'assertion est vraie. Supposons le résultat vrai pour $j = k \geq 0$. Posons $(\tau_1^{\aleph})^k(\emptyset) = S_1$. On a

$$(\tau_1^{\aleph})^{k+1}(\emptyset) = \tau_1^{\aleph}((\tau_1^{\aleph})^k(\emptyset)) = \tau_1^{\aleph}(S_1).$$

Ainsi on a

$$s \in (\tau_1^{\aleph})^{k+1}(\emptyset) \Leftrightarrow \mathbf{e}_s(f) > 0 \text{ ou } \mathbf{e}_s(\mathbf{AX}(S_1)) > 0.$$

Si $\mathbf{e}_s(f) > 0$ alors s satisfait la propriété et le problème est réglé. Sinon, remarquons d'abord que $(\tau_1^{\aleph})^k(\emptyset)$ est l'ensemble des éléments de $s \in S$ satisfaisant la condition que pour toute exécution π de longueur k issue de s, il y a au moins un entier $j_s \leq k$ tel que $\mathbf{e}_{\pi[j_s]}(f) > 0$. Donc si $\mathbf{e}_s(f) = 0$ on aura $\mathbf{e}_s(\mathbf{AX}(S_1)) > 0$ et tous les successeurs de s satisfont la propriété que toutes les exécutions de longueur k qui en sont issues contiennent un état où f est satisfaite. Donc toutes les exécutions de longueur $k+1$ issues de s satisfont la même propriété. Le résultat souhaité est démontré.

Lemme 6 (Propriétés des itérées sur S) *Soient $i \in \{3,\ 4,\ 7,\ 8\}$, alors la jème itérée des transformations de prédicats $(\tau_i^{\aleph})^j$ peut être décrite ainsi:*

1. *$(\tau_3^{\aleph})^j(S)$ est le sous-ensemble des éléments x de S tels que pour toute exécution π de longueur j issue de x, on a que tous les états $\pi[k], 0 \leq k \leq j$ de l'exécution satisfont $\mathbf{e}_{\pi[k]}(f) > 0$.*

2. *$(\tau_4^{\aleph})^j(S)$ est le sous-ensemble des éléments x de S tels qu'il existe une exécution π de longueur j issue de x satisfaisant, pour tout entier $k, 0 \leq k \leq j$ la condition $\mathbf{e}_{\pi[k]}(f) > 0$.*

3. *$(\tau_7^{\aleph})^j(S)$ est les sous-ensemble des éléments x de S tels que pour tout chemin π issu de x de longueur j doit satisfaire la condition $\mathbf{e}_{\pi[k]}(g) > 0, 0 \leq k \leq j$, jusqu'à ce que les conditions $\mathbf{e}_{\pi[m]}(f) > 0$, et $\mathbf{e}_{\pi[m]}(g) > 0,\ 0 \leq m \leq j$ soient vraies toutes deux.*

4. *$(\tau_8^{\aleph})^j(S)$ est les sous-ensemble des éléments x de S tels qu'il existe un chemin π issu de x de longueur j qui doit satisfaire la condition $\mathbf{e}_{\pi[k]}(g) > 0, 0 \leq k \leq j$, jusqu'à ce que les conditions $\mathbf{e}_{\pi[m]}(f) > 0$, et $\mathbf{e}_{\pi[m]}(g) > 0,\ 0 \leq m \leq j$ soient vraies toutes deux.*

Nous allons montrer dans la proposition suivante que toutes ces transformations sont aussi monotones, au sens classique du terme. N'oublions pas que l'ensemble des états n'est pas flou. Ce sont les critères d'appartenance qui le sont. Il n'y a donc pas lieu de généraliser ici la notion de monotonicité au cas flou et le théorème de Knaster-Tarski s'applique directement.

Proposition 7 (Monotonicité) *Pour toute structure de Kripke floue $K_{\aleph}$, tous sous-ensembles A, B de l'ensemble des états de $K_{\aleph}$ tels que $A \subseteq B$ et tout entier i tel que $1 \leq i \leq 8$, on a*

$$A \subseteq B \ \Rightarrow\ \tau_i^{\aleph}(A) \subseteq \tau_i^{\aleph}(B).$$

Démonstration.

1. Soit $s \in \tau_1^{\aleph}(A) = f \vee \mathbf{AX}A$. Alors on a

$$\mathrm{Max}(\mathbf{e}_s(f),\ \mathbf{e}_s(\mathbf{AX}A)) > 0.$$

Si $\mathbf{e}_s(f) > 0$ alors s est dans $s \in \tau_1^{\aleph}(B)$, par définition même de $\tau_1^{\aleph}$. Sinon on doit avoir $\mathbf{e}_s(\mathbf{AX}A) > 0$ qui signifie que pour tout état s' de $K_{\aleph}$ tel que $R_{\aleph}(s, s') > 0$ on a $\mathbf{e}_{s'}(A) > 0$, indiquant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_1^{\aleph}(B)$.

2. Soit $s \in \tau_2^{\aleph}(A) = f \vee \mathbf{EX}A$. Alors on a

$$\mathrm{Max}(\mathbf{e}_s(f),\ \mathbf{e}_s(\mathbf{EX}A)) > 0.$$

Si $\mathbf{e}_s(f) > 0$ alors s est dans $s \in \tau_2^{\aleph}(B)$, par définition même de $\tau_2^{\aleph}$. Sinon on doit avoir $\mathbf{e}_s(\mathbf{EX}A) > 0$ qui signifie qu'il existe un état s' de $K_{\aleph}$ tel que $R_{\aleph}(s, s') > 0$ et pour lequel on a $\mathbf{e}_{s'}(A) > 0$, signifiant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_2^{\aleph}(B)$.

3. Soit $s \in \tau_3^{\aleph}(A) = f \wedge \mathbf{AX}A$. Alors on a

$$\mathrm{Min}(\mathbf{e}_s(f),\ \mathbf{e}_s(\mathbf{AX}A)) > 0.$$

Cela entraîne $\mathbf{e}_s(f) > 0$ et $\mathbf{e}_s(\mathbf{AX}A) > 0$. La dernière condition signifie que pour tout état s' de $K_{\aleph}$ tel que $R_{\aleph}(s, s') > 0$ on a $\mathbf{e}_{s'}(A) > 0$, indiquant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi, avec la condition $\mathbf{e}_s(f) > 0$, on a bien $s \in \tau_3(B)$.

4. Soit $s \in \tau_4^{\aleph}(A) = f \wedge \mathbf{EX}A$. Alors on a

$$\mathrm{Min}(\mathbf{e}_s(f), \mathbf{e}_s(\mathbf{EX}A)) > 0.$$

Cela entraîne $\mathbf{e}_s(f) > 0$ et $\mathbf{e}_s(\mathbf{EX}A) > 0$. La dernière condition signifie qu'il existe un état s' de $K_{\aleph}$ tel que $R_{\aleph}(s, s') > 0$ pour lequel on a $\mathbf{e}_{s'}(A) > 0$, indiquant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi , avec la condition $\mathbf{e}_s(f) > 0$, on a bien $s \in \tau_4^{\aleph}(B)$.

5. Soit $s \in \tau_5^{\aleph}(A) = g \vee (f \wedge \mathbf{AX}A)$. Alors on a

$$\mathrm{Max}(\mathbf{e}_s(g),\ \mathbf{e}_s(f \wedge \mathbf{AX}A)) > 0.$$

Si $\mathbf{e}_s(g) > 0$ alors s est dans $s \in \tau_5^{\aleph}(B)$, par définition même de $\tau_5^{\aleph}$. Sinon on doit avoir

$$\mathbf{e}_s(f \wedge \mathbf{AX}A) > 0$$

qui signifie $\mathbf{e}_s(f) > 0$ et $\mathbf{e}_s(\mathbf{AX}A) > 0$. Si tel est le cas alors, pour tout état s' de $K_\aleph$ tel que $R_\aleph(s, s') > 0$, on a $\mathbf{e}_{s'}(A) > 0$, signifiant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_5^\aleph(B)$, car $\mathbf{e}_s(f) > 0$.

6. Soit $s \in \tau_6^\aleph(A) = g \vee (f \wedge \mathbf{EX}A)$. Alors on a

$$\mathrm{Max}(\mathbf{e}_s(g),\ \mathbf{e}_s(f \wedge \mathbf{EX}A)) > 0.$$

Si $\mathbf{e}_s(g) > 0$ alors s est dans $s \in \tau_6^\aleph(B)$, par définition même de $\tau_6^\aleph$. Sinon on doit avoir

$$\mathbf{e}_s(f \wedge \mathbf{EX}A) > 0$$

qui signifie $\mathbf{e}_s(f) > 0$ et $\mathbf{e}_s(\mathbf{EX}A) > 0$. Donc, il existe un état s' de $K_\aleph$ tel que $R_\aleph(s, s') > 0$, pour lequel on a $\mathbf{e}_{s'}(A) > 0$, signifiant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_6^\aleph(B)$ car $\mathbf{e}_s(f) > 0$.

7. Soit $s \in \tau_7^\aleph(A) = g \wedge (f \vee \mathbf{AX}A)$. Alors on a

$$\mathrm{Min}(\mathbf{e}_s(g),\ \mathbf{e}_s(f \vee \mathbf{AX}A)) > 0.$$

Cela entraîne $\mathbf{e}_s(g) > 0$ et $\mathbf{e}_s(f \vee \mathbf{AX}A)) > 0$. La dernière condition signifie $\mathrm{Max}(\mathbf{e}_s(f),\ \mathbf{e}_s(\mathbf{AX}A))$ > 0. Si $\mathbf{e}_s(f) > 0$ alors on a $s \in \tau_7^\aleph(B)$. Sinon, pour tout état s' de $K_\aleph$ tel que $R_\aleph(s, s') > 0$, on a $\mathbf{e}_{s'}(A) > 0$, signifiant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_7^\aleph(B)$.

8. Soit $s \in \tau_8^\aleph(A) = g \wedge (f \vee \mathbf{EX}A)$. Alors on a

$$\mathrm{Min}(\mathbf{e}_s(g),\ \mathbf{e}_s(f \vee \mathbf{EX}A)) > 0.$$

Cela entraîne $\mathbf{e}_s(g) > 0$ et $\mathbf{e}_s(f \vee \mathbf{EX}A)) > 0$. La dernière condition signifie $\mathrm{Max}(\mathbf{e}_s(f),\ \mathbf{e}_s(\mathbf{EX}A))$ > 0. Si $\mathbf{e}_s(f) > 0$ alors on a $s \in \tau_8^\aleph(B)$. Sinon, il existe un état s' de $K_\aleph$ tel que $R_\aleph(s, s') > 0$, pour lequel on a $\mathbf{e}_{s'}(A) > 0$, signifiant que $s' \in A$. Puisque, par hypothèse, $A \subseteq B$, on a $s' \in B$. Ainsi $s \in \tau_8^\aleph(B)$.

Lemme 7 (Limite des itérées)

1. *Soit* $(\tau_1^\aleph)^{n_1}(\emptyset)$ *la limite de la transformation* $\tau_1^\aleph(Z) = f \vee \mathbf{AX}Z$ *alors pour tout* s *dans* S, *si* $s \in (\tau_1^\aleph)^{n_1}(\emptyset)$ *alors on a* $\mathbf{e}_s(f) > 0$ *ou, pour tout* $s' \in S$ *tel que* $R_\aleph(s, s') > 0$, *on a* $\mathbf{e}_{s'}((\tau_1^\aleph)^{n_1}(\emptyset)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{AF}f) > 0$.

2. *Soit* $(\tau_2^{\aleph})^{n_2}(\emptyset)$ *la limite de la transformation* $\tau_2^{\aleph}(Z) = f \vee \mathbf{EX}Z$ *alors pour tout* s *dans* S, *si* $s \in (\tau_2^{\aleph})^{n_2}(\emptyset)$ *alors on a* $\mathbf{e}_s(f) > 0$ *ou il existe* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$, *on a* $\mathbf{e}_{s'}((\tau_2^{\aleph})^{n_2}(\emptyset)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{EF}f) > 0$.

3. *Soit* $(\tau_3^{\aleph})^{n_3}(S)$ *la limite de la transformation* $\tau_3^{\aleph}(Z) = f \wedge \mathbf{AX}Z$ *alors pour tout* s *dans* S, *si* $s \in (\tau_3^{\aleph})^{n_3}(S)$ *alors on a* $\mathbf{e}_s(f) > 0$ *et pour tout* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$ *on a* $\mathbf{e}_{s'}((\tau_3^{\aleph})^{n_3}(S)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{AG}f) > 0$.

4. *Soit* $(\tau_4^{\aleph})^{n_4}(S)$ *la limite de la transformation* $\tau_4^{\aleph}(Z) = f \wedge \mathbf{EX}Z$ *alors pour tout* s *dans* S, *si* $s \in (\tau_4^{\aleph})^{n_4}(S)$ *alors on a* $\mathbf{e}_s(f) > 0$ *et il existe* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$ *pour lequel on a* $\mathbf{e}_{s'}((\tau_4^{\aleph})^{n_4}(S)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{EG}f) > 0$.

5. *Soit* $(\tau_5^{\aleph})^{n_5}(\emptyset)$ *la limite de la transformation* $\tau_5^{\aleph}(Z) = g \vee (f \wedge \mathbf{AX}Z)$ *alors pour tout* s *dans* S, *si* $s \in (\tau_5^{\aleph})^{n_5}(\emptyset)$ *alors on a* $\mathbf{e}_s(g) > 0$ *ou on a* $\mathbf{e}_s(f) > 0$ *et pour tout* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$, *on a* $\mathbf{e}_{s'}((\tau_5^{\aleph})^{n_5}(\emptyset)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{A}[f\mathbf{U}g]) > 0$.

6. *Soit* $(\tau_6^{\aleph})^{n_6}(\emptyset)$ *la limite de la transformation* $\tau_6^{\aleph}(Z) = g \vee (f \wedge \mathbf{EX}Z)$ *alors pour tout* s *dans* S, *si* $s \in (\tau_6^{\aleph})^{n_6}(\emptyset)$ *alors on a* $\mathbf{e}_s(g) > 0$ *ou on a* $\mathbf{e}_s(f) > 0$ *et il existe* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$ *pour lequel on* $\mathbf{e}_{s'}((\tau_6^{\aleph})^{n_6}(\emptyset)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{E}[f\mathbf{U}g]) > 0$.

7. *Soit* $(\tau_7^{\aleph})^{n_7}(S)$ *la limite de la transformation* $\tau_7^{\aleph}(Z) = g \wedge (f \vee \mathbf{AX}Z)$ *alors pour tout* s *dans* S, *si* $s \in (\tau_7^{\aleph})^{n_7}(S)$ *alors on a* $\mathbf{e}_s(g) > 0$ *et pour tout* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$ *on a soit* $\mathbf{e}_s(f) > 0$, *soit* $\mathbf{e}_{s'}((\tau_7^{\aleph})^{n_7}(S)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{A}[f\mathbf{R}g]) > 0$.

8. *Soit* $(\tau_8^{\aleph})^{n_8}(S)$ *la limite de la transformation* $\tau_8^{\aleph}(Z) = g \wedge (f \vee \mathbf{EX}Z)$ *alors pour tout* s *dans* S, *si* $s \in (\tau_8^{\aleph})^{n_8}(S)$ *alors on a* $\mathbf{e}_s(f) > 0$ *et il existe* $s' \in S$ *tel que* $R_{\aleph}(s, s') > 0$ *on a soit* $\mathbf{e}_s(f) > 0$, *soit* $\mathbf{e}_{s'}((\tau_8^{\aleph})^{n_8}(S)) > 0$. *En d'autres termes, on a* $\mathbf{e}_s(\mathbf{E}[f\mathbf{R}g]) > 0$.

Démonstration.

1. Si $\mathbf{e}_s((\tau_1^{\aleph})^{n_1}(\emptyset)) > 0$ alors $\mathbf{e}_s(\tau_1^{\aleph}((\tau_1^{\aleph})^{n_1}(\emptyset))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_1^{\aleph}$ on a $\mathbf{e}_s(f) > 0$ ou alors pour tout $s' \in S$ tel que $R_{\aleph}(s, s') > 0$ on a $s' \in (\tau_1^{\aleph})^{n_1}(\emptyset)$. Si $\mathbf{e}_s(f) > 0$ alors $\mathbf{e}_s(\mathbf{AF}f) > 0$. Sinon, tout s' tel que $R_{\aleph}(s, s') > 0$ se retrouve dans la même situation que l'était s au départ et ainsi pour toute exécution π débutant à l'état s il y a un index j_π, qui en général dépend de l'exécution π , tel que $\mathbf{e}_{\pi[j_\pi]}(f) > 0$. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{AF}f) > 0$.

2. Si $\mathbf{e}_s((\tau_2^{\aleph})^{n_2}(\emptyset)) > 0$ alors $\mathbf{e}_s(\tau_2^{\aleph}((\tau_2^{\aleph})^{n_2}(\emptyset))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_2^{\aleph}$ on a $\mathbf{e}_s(f) > 0$ ou alors il existe $s' \in S$ tel que $R_{\aleph}(s, s') > 0$

pour lequel on a $s' \in (\tau_2^{\aleph})^{n_2}(\emptyset)$. Si $\mathbf{e}_s(f) > 0$ alors $\mathbf{e}_s(\mathbf{EF}f) > 0$. Sinon, il existe s' tel que $R_{\aleph}(s, s') > 0$ qui se retrouve dans la même situation que l'était s au départ et ainsi pour il existe une exécution π débutant à l'état s et un index j_π, qui en général dépend de l'exécution π , tels que $\mathbf{e}_{\pi[j_\pi]}(f) > 0$. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{EF}f) > 0$.

3. Si $\mathbf{e}_s((\tau_3^{\aleph})^{n_3}(S)) > 0$ alors $\mathbf{e}_s(\tau_3^{\aleph}((\tau_3^{\aleph})^{n_3}(S))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_3^{\aleph}$ on a $\mathbf{e}_s(f) > 0$ et pour tout $s' \in S$ tel que $R_{\aleph}(s, s') > 0$ on a $s' \in (\tau_3^{\aleph})^{n_3}(S)$. Clairement, s' se retrouve dans la même situation que l'était s au départ et ainsi on a que toute exécution π débutant à l'état s est telle que pour tout index $j \geq 0$ et tout état $\pi[j]$ la propriété f est telle que $\mathbf{e}_{\pi[j]}(f) > 0$. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{AG}f) > 0$.

4. Si $\mathbf{e}_s((\tau_4^{\aleph})^{n_4}(S)) > 0$ alors $\mathbf{e}_s(\tau_4^{\aleph}((\tau_4^{\aleph})^{n_4}(S))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_4^{\aleph}$ on a $\mathbf{e}_s(f) > 0$ et il existe $s' \in S$ tel que $R_{\aleph}(s, s') > 0$ et $s' \in (\tau_4^{\aleph})^{n_4}(S)$. Clairement, s' se retrouve dans la même situation que l'était s au départ et ainsi il existe une exécution π débutant à l'état s telle que pour tout index $j \geq 0$ et pour tout état $\pi[j]$ la propriété f est telle que $\mathbf{e}_{\pi[j]}(f) > 0$. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{EG}f) > 0$.

7. Si $\mathbf{e}_s((\tau_7^{\aleph})^{n_7}(S)) > 0$ alors $\mathbf{e}_s(\tau_7^{\aleph}((\tau_7^{\aleph})^{n_7}(S))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_7^{\aleph}$ on a $\mathbf{e}_s(g) > 0$ et soit $\mathbf{e}_s(f) > 0$ soit pour tout $s' \in S$ tel que $R_{\aleph}(s, s') > 0$ on a $s' \in \tau_7^{\aleph})^{n_7}(S)$. Dans le dernier cas, clairement, s' se retrouve dans la même situation que l'était s au départ. Ainsi on a que pour toute exécution π débutant à l'état s on doit avoir $\mathbf{e}_{\pi[j]}(g) > 0, j \geq 0$ pour tout état de π jusqu'à ce que la propriété f soit telle que $\mathbf{e}_{\pi[k]}(f) > 0$, si cela arrive un jour. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{A}[f\mathbf{R}g]) > 0$.

8. Si $\mathbf{e}_s((\tau_8^{\aleph})^{n_8}(S)) > 0$ alors $\mathbf{e}_s(\tau_8^{\aleph}((\tau_8^{\aleph})^{n_8}(S))) > 0$ car c'est le point fixe de la transformation. Donc, par définition de $\tau_8^{\aleph}$ on a $\mathbf{e}_s(g) > 0$ et soit $\mathbf{e}_s(f) > 0$ soit il existe $s' \in S$ tel que $R_{\aleph}(s, s') > 0$ tel que $s' \in (\tau_8^{\aleph})^{n_8}(S))$. Dans le dernier cas, clairement, s' se retrouve dans la même situation que l'était s au départ. Ainsi on a que pour qu'il existe une exécution π débutant à l'état s pour laquelle on doit avoir $\mathbf{e}_{\pi[j]}(g) > 0, j \geq 0$ pour tout état de π jusqu'à ce que la propriété f soit telle que $\mathbf{e}_{\pi[k]}(f) > 0$, si cela arrive un jour. Cela qui revient à dire que $\mathbf{e}_s(\mathbf{E}[f\mathbf{R}g]) > 0$. Ceci achève la démonstration.

Nous voulons maintenant calculer les point fixes des transformations que nous venons de présenter. Pour ce faire, la méthode utilisée dans le cas rigide est toujours bonne. Cependant, dans certains cas, on peut aussi procéder en établissant l'égalité *numérique* entre deux évaluations de valeurs de vérités floues plutôt que de montrer l'identité *ensembliste*, comme cela est fait dans le cas normal.

Dit autrement, si $MG(i)$ et $MD(i)$ représentent respectivement les membres gauches et droits des huit identités de la proposition 8,

$$MG(i) \equiv MD(i), 1 \leq i \leq 8$$

alors, pour tout état s de $K_{\aleph}$ et pour tout $i \in \{2,3\}$, on a

$$\mathbf{e}_s(MG(i)) = \mathbf{e}_s(MD(i)),$$

et, pour tout $i \in \{1,4,5,6,7,8\}$, on a

$$\mathbf{e}_s(MG(i)) = 0 \Leftrightarrow \mathbf{e}_s(MD(i)) = 0.$$

Remarquons que l'identité numérique des points 2 et 3 entraîne évidemment l'identité ensembliste de ces mêmes cas.

Proposition 8 (Règle d'expansion des opérateurs $\aleph CTL$) *Pour toute structure de Kripke floue $K_{\aleph}$ et toutes formules de chemins f, g sur $K_{\aleph}$ on a*

1. $\mathbf{AF}g \equiv g \vee \mathbf{AX}\ \mathbf{AF}g$
2. $\mathbf{EF}g \equiv g \vee \mathbf{EX}\ \mathbf{EF}g$
3. $\mathbf{AG}g \equiv g \wedge \mathbf{AX}\ \mathbf{AG}g$
4. $\mathbf{EG}g \equiv g \wedge \mathbf{EX}\ \mathbf{EG}g$
5. $\mathbf{A}(f\ \mathbf{U}g) \equiv g \vee (f \wedge \mathbf{AX}\ \mathbf{A}(f\ \mathbf{U}g))$
6. $\mathbf{E}(f\ \mathbf{U}g) \equiv g \vee (f \wedge \mathbf{EX}\ \mathbf{E}(f\ \mathbf{U}g))$
7. $\mathbf{A}(f\ \mathbf{R}g) \equiv g \wedge (f \vee \mathbf{AX}\ \mathbf{A}(f\ \mathbf{R}g))$
8. $\mathbf{E}(f\ \mathbf{R}g) \equiv g \wedge (f \vee \mathbf{EX}\ \mathbf{E}(f\ \mathbf{R}g))$

où pour toute formule Φ_1 et Φ_2, on a $\Phi_1 \equiv \Phi_2$ si, et seulement si, pour tout état s de l'ensemble S des états de $K_{\aleph}$, on a

$$\mathbf{e}_s(\Phi_1) = 0 \Leftrightarrow \mathbf{e}_s(\Phi_2) = 0.$$

Démonstration. Soit s un état de $K_{\aleph}$.

1. On a

$$\mathbf{e}_s\mathbf{AF}(g)$$

$$= \min_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \mathbf{e}_\pi(\mathbf{F}g)$$

$$= \min_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

$$\leq \mathrm{Max}(\mathbf{e}_s(g),\ (\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}^*} \mathbf{e}_{\pi^j}(g)))$$

$$= \mathrm{Max}(\mathbf{e}_s(g),\ \mathbf{e}_s(\mathbf{AX}\ \mathbf{AF}(g)))$$

$$= \mathbf{e}_s(g) \vee\ \mathbf{e}_s(\mathbf{AX}\ \mathbf{AF}(g))$$

La notation $\mathbf{N}^*$ est utilisée pour désigner l'ensemble des entiers strictement positifs, $\mathbf{N}^* = \{n \in \mathbf{N} | n > 0\}$. Maintenant, pour montrer l'identité souhaitée, il suffit de montrer

$$\mathbf{e}_s(g) \vee\ \mathbf{e}_s(\mathbf{AX}\ \mathbf{AF}(g)) = 0 \Rightarrow \mathbf{e}_s\mathbf{AF}(g) = 0.$$

Par hypothèse, on a

$$\mathrm{Max}(\mathbf{e}_s(g),\ (\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}^*} \mathbf{e}_{\pi^j}(g))) = 0 \Leftrightarrow$$

$$\mathbf{e}_s(g) = 0 \quad \text{et} \quad (\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}^*} \mathbf{e}_{\pi^j}(g)) = 0.$$

La valeur de g en s doit être nulle et il existe au moins un chemin π_0 issu de s sur lequel la valeur de g maximale atteinte (abstraction faite de la racine s) est nulle. En combinant ces deux résultats, on a que la valeur de g en chacun des sommets de π_0, racine comprise, est nulle aussi. Cela signifie que $\mathbf{e}_s\mathbf{AF}(g) = 0$. Le résultat est démontré.

2. $\mathbf{e}_s\mathbf{EF}(g)$

$$= \max_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \mathbf{e}_\pi(\mathbf{F}g)$$

$$= \max_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

$$= \mathrm{Max}(\mathbf{e}_s(g),\ (\max_{\pi\in\mu(K_{\aleph}),\pi[0]=s} \max_{j\in\mathbf{N}^*} \mathbf{e}_{\pi^j}(g)))$$

$$= \mathrm{Max}(\mathbf{e}_s(g),\ \mathbf{e}_s(\mathbf{EX}\ \mathbf{EF}(g)))$$

$$= \mathbf{e}_s(g) \vee\ \mathbf{e}_s(\mathbf{EX}\ \mathbf{EF}(g))$$

3. $\mathbf{e}_s\mathbf{AG}(g)$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(\mathbf{G}g)$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

$$= \mathrm{Min}(\mathbf{e}_s(g),\ (\min_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}^\bullet} \mathbf{e}_{\pi^j}(g)))$$

$$= \mathrm{Min}(\mathbf{e}_s(g),\ \mathbf{e}_s(\mathbf{AX}\ \mathbf{AG}(g)))$$

$$= \mathbf{e}_s(g) \wedge\ \mathbf{e}_s(\mathbf{AX}\ \mathbf{AG}(g))$$

4. $\mathbf{e}_s\mathbf{EG}(g)$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(\mathbf{G}g)$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(g)$$

$$\geq \mathrm{Min}(\mathbf{e}_s(g),\ \max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\min_{j\in\mathbf{N}^\bullet} \mathbf{e}_{\pi^j}(g))$$

$$= \mathrm{Min}(\mathbf{e}_s(g),\ \mathbf{e}_s(\mathbf{EX}\ \mathbf{EG}(g)))$$

$$= \mathbf{e}_s(g) \wedge\ \mathbf{e}_s(\mathbf{EX}\ \mathbf{EG}(g))$$

Ici on doit montrer

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(g) = 0 \Rightarrow \mathrm{Min}(\mathbf{e}_s(g),\ (\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}^\bullet} \mathbf{e}_{\pi^j}(g))) = 0$$

Par hypothèse, on a que pour tout chemin π issu de s, il existe au moins un sommet de π où la valeur de g est nulle. Ce sommet peut être s lui-même. Si ce n'est pas le cas il doit alors se trouver sur un des successeurs de s dans π. C'est ce qu'indique la formule

$$\mathrm{Min}(\mathbf{e}_s(g),\ (\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{j\in\mathbf{N}^\bullet} \mathbf{e}_{\pi^j}(g))) = 0.$$

Le résultat est démontré

5. Nous allons montrer que

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{U}g) = 0 \Leftrightarrow \mathbf{e}_s(g\ \vee\ (f\ \wedge \mathbf{AX}\ \mathbf{A}(f\ \mathbf{U}g))) = 0.$$

($\Rightarrow$). On a

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{U}g) = \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(f\mathbf{U}g) = 0$$

si et seulement si, par définition,

$$\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}\max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f)))=0.$$

Il doit donc y avoir au moins un chemin π_0 issu de s tel que

$$\max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi_0^k}(g),\min_{0\leq j<k}\ \mathbf{e}_{\pi_0^j}(f)))=0.$$

Sur π_0, pour tout entier $k\geq 0$ on a alors

$$\mathrm{Min}(\ \mathbf{e}_{\pi_0^k}(g),\min_{0\leq j<k}\ \mathbf{e}_{\pi_0^j}(f))=0.$$

Lorsque $k=0$ on a

$$\mathrm{Min}(\ \mathbf{e}_{\pi_0^0}(g),\min_{0\leq j<0}\ \mathbf{e}_{\pi_0^j}(f))=\mathrm{Min}(\ \mathbf{e}_{\pi_0^0}(g),1)=\mathbf{e}_{\pi_0^0}(g)=0.$$

Deux possibilités se présentent alors:

A. pour tout entier $k\geq 1$ on a $\mathbf{e}_{\pi_0^k}(g)=0$,

B. il y a au moins un entier $k\geq 1$ tel que $\mathbf{e}_{\pi_0^k}(g)\neq 0$ et alors $\min_{0\leq j<k}\ \mathbf{e}_{\pi_0^j}(f))=0$.

Dans un cas comme dans l'autre, l'expression

$$\mathbf{e}_s(g\ \vee\ (f\ \wedge\mathbf{AX}\ \mathbf{A}(f\ \mathbf{U}g)))=$$

$$\mathrm{Max}(\mathbf{e}_{\pi^0}(g),\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))))=$$

$$\mathrm{Max}(0,\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))))=$$

$$\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))$$

est nulle. En effet si l'hypothèse A est vraie alors on a

$$\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ 0,\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))=0,$$

et si l'hypothèse B est vraie alors on a plutôt

$$\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),0))=0.$$

Le critère nécessaire est démontré.

($\Leftarrow$). Si maintenant on suppose vraie la formule

$$\mathbf{e}_s(g\ \vee\ (f\ \wedge\mathbf{AX}\ \mathbf{A}(f\ \mathbf{U}g))=0$$

alors, pour tout chemin π issu de S, les deux identités suivantes

$$\mathbf{e}_{\pi^0}(g)=0\quad\text{et}\quad\min(\mathbf{e}_{\pi^0}(f),\min_{\pi\in\mu(K_{\aleph}),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))=0$$

sont vraies. La deuxième de ces identités entraîne que, pour tout chemin π issu de s, on a

A. $\mathbf{e}_{\pi^0}(f) = 0$ ou

B. $\min_{\pi\in\mu(K_\aleph),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))) = 0.$

Lorsque l'hypothèse A est vérifiée, on a

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{U}g)$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(f\mathbf{U}g)$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f)))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), 0)) = 0.$$

Lorsque l'hypothèse B est vérifiée, il existe au moins un chemin π_0 pour lequel

$$\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi_0^k}(g), \min_{0<j<k}\ \mathbf{e}_{\pi_0^j}(f))) = 0. \qquad (\dagger)$$

On a alors

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{U}g)$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(f\mathbf{U}g)$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f)))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \min_{0\le j<0}\ \mathbf{e}_{\pi^j}(g)), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \mathbf{1}), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathbf{e}_{\pi^0}(g), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathbf{e}_{\pi^0}(g), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \mathrm{Min}(\mathbf{e}_{\pi^0}(f), \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \min_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathbf{e}_{\pi^0}(g), \mathrm{Min}(\mathbf{e}_{\pi^0}(f), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))).$$

Maintenant, lorsqu'on fixe le quantificateur des chemins à $\pi = \pi_0$, la dernière expression se réduit et est égale à ce qui suit en utilisant (†).

$$= \mathrm{Max}(\mathbf{e}_{\pi_0^0}(g), \mathrm{Min}(\mathbf{e}_{\pi_0^0}(f), 0)) = 0, \text{ car } \mathbf{e}_{\pi^0}(g) = 0.$$

Le critère suffisant est démontré.

6. Nous voulons montrer que

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g) = 0 \Leftrightarrow \mathbf{e}_s(g\ \vee\ (f\ \wedge \mathbf{EX}\ \mathbf{E}(f\ \mathbf{U}g))) = 0.$$

($\Rightarrow$). On a

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \mathbf{e}_\pi(f\mathbf{U}g) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f))) = 0$$

Ainsi, pour tout chemin π issu de s et tout entier $k \geq 0$ on a

$$\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f)) = 0$$

ce qui implique que, pour tout entier $k \geq 0$, soit

$$\mathbf{e}_{\pi^k}(g) = 0,$$

soit

$$\min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f) = 0.$$

Si, pour tout entier $k \geq 0$ on a $\mathbf{e}_{\pi^k}(g) = 0$ alors

$$\mathbf{e}_s(g\ \vee\ (f\ \wedge \mathbf{EX}\ \mathbf{E}(f\ \mathbf{U}g)))$$

$$= \mathrm{Max}(\mathbf{e}_{\pi^0}(g), \mathrm{Min}(\mathbf{e}_{\pi^0}(f), \max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \mathrm{Max}(0, \mathrm{Min}(\mathbf{e}_{\pi^0}(f), \max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ 0, \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))$$

$$= \mathrm{Max}(0, \mathrm{Min}(\mathbf{e}_{\pi^0}(f), \max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\max_{k\in\mathbf{N}^*}(0)))$$

$$= \mathrm{Max}(0, \mathrm{Min}(\mathbf{e}_{\pi^0}(f), 0)) = 0$$

Par ailleurs on a

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f))) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \min_{0\le j<0}\ \mathbf{e}_{\pi^j}(g)), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f)))) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_\aleph),\pi[0]=s} (\mathrm{Max}(\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \mathbf{1}), \max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\le j<k}\ \mathbf{e}_{\pi^j}(f)))) = 0 \Leftrightarrow$$

$$\max_{\pi\in\mu(K_N),\pi[0]=s}(\mathrm{Max}(\mathbf{e}_{\pi^0}(g),\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f))))=0$$

Cela entraîne que $\mathbf{e}_{\pi^0}(g)=0$, car sinon l'expression précédente n'est pas nulle. Supposons maintenant π un chemin fixé issu de s. Supposons aussi que l'on ait, pour un certain entier $r>0$, $\mathbf{e}_{\pi^r}(g)\neq 0$. Alors on a

$$\min_{0\leq j<r}\ \mathbf{e}_{\pi^j}(f)=0.$$

Il existe donc un entier $t, 0\leq t<r$ tel que $\mathbf{e}_{\pi^t}(f)=0$. Sans perte de généralité on peut aussi supposer que pour tout entier i tel que $0\leq i<r$ on a $\mathbf{e}_{\pi^i}(g)=0$, c'est-à-dire que r est le plus petit entier tel que $\mathbf{e}_{\pi^r}(g)\neq 0$. Dans ce cas on obtient

$$\begin{aligned}
&\mathbf{e}_s(g\ \vee\ (f\ \wedge\mathbf{EX}\ \mathbf{E}(f\ \mathbf{U}g)))\\
&=\mathrm{Max}(\mathbf{e}_{\pi^0}(g),\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\max_{\pi\in\mu(K_N),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))\\
&=\mathrm{Max}(0,\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\max_{\pi\in\mu(K_N),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))\\
&=\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\max_{\pi\in\mu(K_N),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))
\end{aligned}$$

Si on a $\mathbf{e}_{\pi^0}(f)=0$ alors la dernière expression est nulle et le résultat démontré. Sinon, on sait qu'il existe un entier $t, 0<t<r$ tel que $\mathbf{e}_{\pi^t}(f)=0$. Mézalors l'expression

$$\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))$$

est nulle car au moins un des deux termes $\mathbf{e}_{\pi^k}(g)$ ou $\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)$ l'est aussi. Ce résultat étant valide pour tout chemin π issu de s, le résultat est démontré.

($\Leftarrow$). Par définition, on a

$$\mathbf{e}_s(g\ \vee\ (f\ \wedge\mathbf{EX}\ \mathbf{E}(f\ \mathbf{U}g)))=0\Leftrightarrow$$

$$\mathrm{Max}(\mathbf{e}_{\pi^0}(g),\mathrm{Min}(\mathbf{e}_{\pi^0}(f),\max_{\pi\in\mu(K_N),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))=0.$$

On a donc, par hypoyhèse, $\mathbf{e}_{\pi^0}(g)=0$ et soit

$$\mathbf{e}_{\pi^0}(f)=0,$$

soit

$$\max_{\pi\in\mu(K_N),\pi[0]=s}(\max_{k\in\mathbf{N}^*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g),\min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))=0.$$

Dans le premier cas, lorsque $\mathbf{e}_{\pi^0}(f)=0$, on a

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g)$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f)))$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), 0))$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(0) = 0.$$

Le résultat est démontré.

Dans le second cas, on a, pour tout chemin π issu de s et tout entier $k > 0$,

$$\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0<j<k}\ \mathbf{e}_{\pi^j}(f)) = 0.$$

Si, pour tout $k > 0$ on a $\mathbf{e}_{\pi^k}(g) = 0$ alors, puisque $\mathbf{e}_{\pi^0}(g) = 0$ on a

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g)$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ 0, \min_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f)))$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(0)) = 0$$

Le résultat est démontré.

S'il existe $k > 0$ tel que $\mathbf{e}_{\pi^k}(g) \neq 0$ alors il existe $j, 0 < j < k$ tel que $\mathbf{e}_{\pi^j}(f) = 0$. Ainsi on a

$$\mathbf{e}_s\mathbf{E}(f\ \mathbf{U}g)$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), 0))$$

$$= \max_{\pi\in\mu(K_\aleph),\pi[0]=s} \max_{k\in\mathbf{N}}(0) = 0.$$

Le résultat est démontré entièrement.

7. Nous voulons montrer que

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{R}g) = 0 \Leftrightarrow \mathbf{e}_s(g\ \wedge\ (f\ \vee \mathbf{AX}\ \mathbf{A}(f\ \mathbf{R}g))) = 0.$$

($\Rightarrow$). Par définition, on a

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{R}g) = 0 \Leftrightarrow \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{k\in\mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f))) = 0.$$

Cette dernière condition n'est vraie que s'il existe au moins un chemion π issu de s et au moins un entier $k \geq 0$ tels que

$$\mathbf{e}_{\pi^k}(g) = 0 \quad \text{et} \quad \max_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f)) = 0.$$

Si c'est le cas, la dernière égalité implique, pour tout $j, 0 \leq j < k, \mathbf{e}_{\pi^j}(f) = 0$.

Avec ces hypothèses, on veut montrer que $\mathbf{e}_s(g \wedge (f \vee \mathbf{AX}\ \mathbf{A}(f\ \mathbf{R}g))) = 0$ c'est-à-dire que

$$\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \mathrm{Max}(\mathbf{e}_{\pi^0}(f), \min_{\pi \in \mu(K_\aleph), \pi[0]=s} (\min_{k \in \mathbf{N}^*} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))) = 0$$

Si $k = 0$ alors $\mathbf{e}_{\pi^0}(g) = 0$ et le résultat est démontré. Si $k > 0$ clairement on a $\mathbf{e}_{\pi^0}(f) = 0$. De plus, on sait qu'il existe au moins un chemin π et un entier k où

$$\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f)) = 0.$$

Puisqu'on prend le minimum sur tous les chemins issus de s et tous les entiers $k > 0$, on a

$$\mathrm{Max}(\mathbf{e}_{\pi^0}(f), \min_{\pi \in \mu(K_\aleph), \pi[0]=s} (\min_{k \in \mathbf{N}^*} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f))) = 0$$

et le résultat est démontré.

($\Leftarrow$). Par définition, on a

$$\mathbf{e}_s(g \wedge (f \vee \mathbf{AX}\ \mathbf{A}(f\ \mathbf{R}g))) = 0 \Leftrightarrow$$

$$\mathrm{Min}(\mathbf{e}_{\pi^0}(g), \mathrm{Max}(\mathbf{e}_{\pi^0}(f), \min_{\pi \in \mu(K_\aleph), \pi[0]=s} (\min_{k \in \mathbf{N}^*} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f))))) = 0.$$

Notre hypothèse mène à deux cas que nous analysons séparément,

A. $\mathbf{e}_{\pi^0}(g) = 0$ et

B. $\min_{\pi \in \mu(K_\aleph), \pi[0]=s} (\min_{k \in \mathbf{N}^*} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f)))) = 0.$

Pour le cas A, en considérant l'index $k = 0$ dans le second quantificateur min et en notant l'identité

$$\max_{0 \leq j<0}\ \mathbf{e}_{\pi^j}(f) = 0$$

nous avons clairement

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{R}g) = \min_{\pi \in \mu(K_\aleph), \pi[0]=s} \min_{k \in \mathbf{N}} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0 \leq j<k}\ \mathbf{e}_{\pi^j}(f))) = 0.$$

Pour le cas B, on a pour hypothèses

$$\mathbf{e}_{\pi^0}(f) = 0$$

et

$$\min_{\pi \in \mu(K_\aleph), \pi[0]=s} \min_{k \in \mathbf{N}^*} (\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0<j<k}\ \mathbf{e}_{\pi^j}(f))) = 0.$$

Cela signifie qu'il existe un chemin π_0 entier $k_0 > 0$ tel que $\mathbf{e}_{\pi_0^{k_0}}(g) = 0$ et tel que pour tout $j, 0 < j < k_0$, on a $\mathbf{e}_{\pi_0^j}(f) = 0$. Ajoutons à cela le fait que $\mathbf{e}_{\pi_0^0}(f) = 0$ et nous obtenons immédiatement le résultat souhaité considérant le chemin π_0 et l'index k_0 dans ce qui suit.

$$\mathbf{e}_s\mathbf{A}(f\ \mathbf{R}g) = \min_{\pi\in\mu(K_\aleph),\pi[0]=s} \min_{k\in\mathbf{N}}(\mathrm{Max}(\ \mathbf{e}_{\pi^k}(g), \max_{0\leq j<k}\ \mathbf{e}_{\pi^j}(f))) = 0.$$

Le résultat est démontré.

8. La preuve du numéro 8 ne diffère pas beaucoup des preuves des identités 5, 6 et 7. Elle est laissée au lecteur.

Proposition 9 (Les plus petits points fixes de la logique $\aleph CTL$) *Pour toute structure de Kripke floue $K_\aleph$ et toutes formules de chemins f, g sur $K_\aleph$ et tout état s de $K_\aleph$ on a:*

1. $\mathbf{AF}f = \bigcup_{j\geq 0}(\tau_1^\aleph)^j(\emptyset)$.
2. $\mathbf{EF}f = \bigcup_{j\geq 0}(\tau_2^\aleph)^j(\emptyset)$.
3. $\mathbf{A}[f\mathbf{U}g] = \bigcup_{j\geq 0}(\tau_5^\aleph)^j(\emptyset)$.
4. $\mathbf{E}[f\mathbf{U}g] = \bigcup_{j\geq 0}(\tau_6^\aleph)^j(\emptyset)$.

Démonstration.

1. Montrons d'abord l'inclusion $\bigcup_{j\geq 0}(\tau_1^\aleph)^j(\emptyset) \subseteq \mathbf{AF}f$. Procédons par récurrence. Clairement, $(\tau_1^\aleph)^0(\emptyset) = \emptyset \subseteq \mathbf{AF}f$. Supposons k un entier tel que $(\tau_1^\aleph)^k(\emptyset) \subseteq \mathbf{AF}f$. Alors, pour tout x dans $(\tau_1^\aleph)^k(\emptyset)$, on a $x \in (\tau_1^\aleph)^{k+1}(\emptyset)$ si et seulement si, on a soit $\mathbf{e}_x(f) > 0$, en quel cas $x \in \mathbf{AF}f$, soit, pour tout état $y \in S$ tel que $R_\aleph(x, y) > 0$, on a $y \in (\tau_1^\aleph)^k(\emptyset)$. Par hypothèse de récurrence, pour tous les états $y \in (\tau_1^\aleph)^k(\emptyset)$, on a $\mathbf{e}_y(\mathbf{AF}(f)) > 0$. Ainsi, par hypothèse de récurrence, on obtient

$$= \mathrm{Max}(\mathbf{e}_x(f), \min_{\pi\in\mu(K_\aleph),\pi[0]=x} \max_{j\in\mathbf{N}*}\mathbf{e}_{\pi^j}(f)) > 0$$

ce qui, par la proposition 8, implique

$$\mathbf{e}_x\mathbf{AF}(f) > 0.$$

Réciproquement, on a vu au lemme 5 que $(\tau_1^\aleph)^j(\emptyset)$ est l'ensemble des éléments $x \in S$ tels que pour toute exécution π de longueur j issue de x, il y a au moins un des états $\pi[k], 0 \leq k \leq j$

tel que $\mathbf{e}_{\pi[k]} > 0$. Soit $m(\pi)$ le minimum de ces entiers, pour chaque exécution π issue de x. Alors il existe un entier m plus grand que le maximum sur tous les chemins issus de x des $m(\pi)$, puisqu'il n'y en a qu'un nombre fini. Clairement, on a $s \in (\tau_1^{\aleph})^m$.

2. Montrons d'abord l'inclusion $\bigcup_{j\geq 0}(\tau_2^{\aleph})^j(\emptyset) \subseteq \mathbf{EF}f$. Procédons par récurrence. Clairement, $(\tau_2^{\aleph})^0(\emptyset) = \emptyset \subseteq \mathbf{EF}f$. Supposons k un entier tel que $(\tau_2^{\aleph})^k(\emptyset) \subseteq \mathbf{EF}f$. Alors, pour tout x dans $(\tau_2^{\aleph})^k(\emptyset)$, on a $x \in (\tau_2^{\aleph})^{k+1}(\emptyset)$ si et seulement si, on a soit $\mathbf{e}_x(f) > 0$, en quel cas $x \in \mathbf{EF}f$, soit il existe un état $y \in S$ tel que $R_{\aleph}(x,y) > 0$, pour lequel on a $y \in (\tau_2^{\aleph})^k(\emptyset)$. Par hypothèse de récurrence, pour cet état $y \in (\tau_2^{\aleph})^k(\emptyset)$, on a $\mathbf{e}_y(\mathbf{EF}(f)) > 0$. Ainsi, on obtient

$\mathbf{e}_x(\mathbf{EF}(f))$

$= \mathrm{Max}_{\pi\in\mu(K_{\aleph}),\pi[0]=x} \max_{j\in\mathbf{N}} \mathbf{e}_{\pi^j}(f)$

$= \max(\mathbf{e}_x(f), \mathrm{Max}_{\pi\in\mu(K_{\aleph}),\pi[0]=x} \max_{j\in\mathbf{N}*} \mathbf{e}_{\pi^j}(f))$

$= \max(\mathbf{e}_x(f), \lambda) > 0$

car, par hypothèse de récurrence λ est un nombre plus grand que zéro.

Réciproquement, on a vu au lemme 5 que $(\tau_2^{\aleph})^j(\emptyset)$ est l'ensemble des éléments $x \in S$ tels qu'il existe une exécution π de longueur j issue de x pour laquelle il y a au moins un des états $\pi[k], 0 \leq k \leq j$, tel que $\mathbf{e}_{\pi[k]} > 0$. Pour chaque exécution π satisfaisant cette condition, soit $m(\pi)$ le minimum de ces entiers. Alors si m est le plus petit des entiers $m(\pi)$ clairement on a $s \in (\tau_2^{\aleph})^m$.

3. Montrons d'abord l'inclusion $\bigcup_{j\geq 0}(\tau_5^{\aleph})^j(\emptyset) \subseteq \mathbf{A}[f\mathbf{U}g]$. Procédons par récurrence. Clairement $(\tau_5^{\aleph})^0(\emptyset) = \emptyset \subseteq \mathbf{A}[f\mathbf{U}g]$. Supposons k un entier tel que $(\tau_5^{\aleph})^k(\emptyset) \subseteq \mathbf{A}[f\mathbf{U}g]$. Alors, pour tout x dans $(\tau_5^{\aleph})^k(\emptyset)$, on a $x \in (\tau_5^{\aleph})^{k+1}(\emptyset)$ si et seulement si, on a soit $\mathbf{e}_x(g) > 0$, en quel cas $x \in \mathbf{A}[f\mathbf{U}g]$, soit $\mathbf{e}_x(f) > 0$ et pour tout état $y \in S$ tel que $R_{\aleph}(x,y) > 0$, on a $y \in (\tau_5^{\aleph})^k(\emptyset)$. Par hypothèse de récurrence, pour tout état $y \in (\tau_5^{\aleph})^k(\emptyset)$, on a $\mathbf{e}_y(\mathbf{A}[f\mathbf{U}g]) > 0$. Ainsi, par hypothèse de récurrence, on obtient

$$= \mathrm{Max}(\mathbf{e}_x(f), \max_{\pi\in\mu(K_{\aleph}),\pi[0]=x} \max_{j\in\mathbf{N}*}(\mathrm{Min}(\ \mathbf{e}_{\pi^k}(g), \min_{0\leq j<k}\ \mathbf{e}_{\pi^k}(f)))$$

ce qui, par la proposition 8, implique

$$\mathbf{e}_x\mathbf{A}[f\mathbf{U}g]) > 0.$$

Réciproquement, on a vu au lemme 5 que $(\tau_5^{\aleph})^j(\emptyset)$ est l'ensemble des éléments $x \in S$ tels qu'il existe une exécution π de longueur j issue de x pour laquelle il y a au moins un des états $\pi[k], 0 \leq k \leq j$, tel que $\mathbf{e}_{\pi[k]} > 0$. Pour chaque exécution π satisfaisant cette condition, soit $m(\pi)$ le minimum de ces entiers. Alors si m est le plus petit des entiers $m(\pi)$ clairement on a $s \in (\tau_5^{\aleph})^m$.

4. La démonstration est similaire aux trois points précédents et est laissée au lecteur.

Proposition 10 (Les plus grands points fixes de la logique $\aleph CTL$) *Pour toute structure de Kripke floue $K_{\aleph}$ et toutes formules de chemins f, g sur $K_{\aleph}$ si S est l'ensemble des états de $K_{\aleph}$ on a:*

1. $\mathbf{AG}f = \bigcap_{j \geq 0} (\tau_3^{\aleph})^j(S)$.
2. $\mathbf{EG}f = \bigcap_{j \geq 0} (\tau_4^{\aleph})^j(S)$.
3. $\mathbf{A}[f\mathbf{R}g] = \bigcap_{j \geq 0} (\tau_7^{\aleph})^j(S)$.
4. $\mathbf{E}[f\mathbf{R}g] = \bigcap_{j \geq 0} (\tau_8^{\aleph})^j(S)$.

Démonstration. Pour alléger la preuve et éviter des redondances trop grandes dans cette preuve, posons $\mathbf{AG} = \mathbf{O_3}$, $\mathbf{EG} = \mathbf{O_4}$, $\mathbf{A[fRg]} = \mathbf{O_7}$, $\mathbf{E[fRg]} = \mathbf{O_8}$ et $I = \{3, 4, 7, 8\}$.

Nous voulons montrer que pour tout $i \in I$, on a

$$\mathbf{O_i}f = \bigcap_j (\tau_i^{\aleph})^j(S).$$

Nous allons procéder en démontrant, pour tout $i \in I$, que l'on a à la fois $\mathbf{O_i}f \subseteq \bigcap_j (\tau_i^{\aleph})^j(S)$ et $\bigcap_j (\tau_i^{\aleph})^j(S) \subseteq \mathbf{O_i}f$.

Pour la première partie, on procède par récurrence. Pour tout $i \in I$ on a $\mathbf{O_i}f \subseteq S$, clairement. Supposons $\mathbf{O_i}f \subseteq (\tau_i^{\aleph})^n(S)$. Par la proposition 7 on a, pour tout $i \in I$, que $(\tau_i^{\aleph})$ est monotone. Donc, pour tout $i \in I$, on a $\tau_i^{\aleph}(\mathbf{O_i}f) \subseteq \tau_i^{\aleph}(\tau_i^{\aleph})^n(S)$. Par le théorème 1 on a, pour tout $i \in I$ que $(\mathbf{O_i}f)$ est un point fixe de $\tau_i^{\aleph}$. D'où le résultat.

Pour l'inclusion réciproque, soit $i \in I$ et $x_i \in S$ tel que $\mathbf{e}_{x_i}(\bigcap_j (\tau_i^{\aleph})^j(S) > 0$. Alors on a, pour tout entier j et tout $i \in I$, $\mathbf{e}_{x_i}(\tau_i^{\aleph})^j(S)) > 0$. Pour chaque $i \in I$ la transformation $\tau_i^{\aleph}$ admet un

point fixe obtenu en itérant $\tau_i^{\aleph}$ sur S un nombre n_i de fois, c'est-à-dire que, pour tout $i \in I$ il existe n_i tel que

$$\tau_i^{\aleph}((\tau_i^{\aleph})^{n_i}(S)) = (\tau_i^{\aleph})^{n_i}(S).$$

Ainsi, pour tout i dans I on a

$$x_i \in \tau_i^{\aleph}((\tau_i^{\aleph})^{n_i}(S)).$$

Pour continuer la preuve à partir de l'équation précédente il faut maintenant distinguer les quatre cas de départ et interpréter les formules dans chaque cas. Il suffit d'appliquer, dans l'ordre, le lemme 7 d'induction à chaque cas pour obtenir le résultat souhaité.

3.5 Conclusion

Dans ce chapitre nous avons défini la logique temporelle floue $\aleph CTL^*$, ce qui nous a permis de définir les opérateurs de la sous-logique $\aleph CTL$. Nous avons ensuite démontré que ces derniers opérateurs étaient tous les points fixes de transformations monotones au sens usuel du terme. Opérant dans un contexte fini, il s'ensuit que l'ensemble des états satisfaisant une formule quelconque exprimée à l'aide des opérateurs $\aleph CTL$ peut être construit récursivement en un nombre fini d'étapes.

CHAPITRE IV

LA VÉRIFICATION DE MODÈLES FLOUE

Abstraitement, la vérification globale de modèles classique consiste à identifier algorithmiquement le sous-ensemble des états d'un système donné qui satisfont une propriété particulière du système.

Il y a plusieurs manières de concrétiser le problème. Par exemple, à l'instar de la méthode suivie dans ce mémoire, on peut décider de représenter le système avec une structure de Kripke et les propriétés avec l'aide de la logique CTL. Plusieurs autres algorithmes de vérification de modèles existent, qui peuvent dépendre de la logique utilisée, bien sûr, mais aussi du type de représentation que l'automate (ou la structure de Kripke) représentant le modèle du système examiné peut avoir.

4.1 La vérification de modèles avec la logique CTL

Plusieurs preuves distinctes de la calculabilité des propriétés CTL sur une structure de Kripke donnée existent. Le traitement peut se faire directement sur le graphe orienté de l'automate représentant la structure de Kripke en examinant les états du système avec l'aide de la relation de transition et en décidant progressivement de ce qu'un état fait ou ne fait pas partie du sous-ensemble des états satisfaisant une propriété donnée. C'est la vérification de modèle classique.

Une autre méthode consiste à utiliser des transformations de prédicats pour construire en un nombre fini d'étapes l'ensemble de tous les états satisfaisant une propriété donnée. Cela est possible car, comme nous l'avons vu, les plus petit et plus grand points fixes de ces transformations s'obtiennent toujours en un nombre fini d'étapes lorsqu'elles sont monotones. Le problème revient alors à montrer leur monotonicité et à bien identifier les transformations associées aux opérations de CTL de manière à les caractériser parfaitement.

Cette dernière méthode est appelée la vérification itérative ou sémantique de modèles. Un de ses avantages vient de ce qu'elle peut traiter des données mises en oeuvre sous forme de diagrammes de décisions binaires (BDD), permettant une gestion extrêmement efficace, relativement à la méthode classique, du calcul des vérifications. On parle alors de vérification *symbolique* de modèles.

De plus, cette méthode a l'avantage de situer le problème de la vérification en termes ensemblistes plutôt qu'en termes de graphes, avantage qui a pour conséquences d'ouvrir grande la porte des généralisations. Voilà pourquoi nous avons opté pour la vérification itérative comme base de notre travail. Ajoutons qu'il n'a absolument pas été nécessaire de faire appel aux diagrammes binaires pour mener à bien notre projet. Les BDD ne sont que des outils de représentation lors de la mise en oeuvre effective. Or nous n'avons examiné que l'*existence* de la vérification floue, et pas sa mise en oeuvre.

4.2 La vérification de modèles «floue»

La vérification de modèles floue que nous présentons ici est une généralisation de la vérification de modèle itérative décrite dans la section précédente. Nous avons généralisé les structures de Kripke pour les rendre floues (dans une certaine mesure), puis nous avons fait de même avec la logique CTL pour pouvoir construire les propriétés de cette nouvelle représentation.

La vérification de modèle floue se définit de la même manière que la vérification itérative ordinaire à un petit détail près. Soient $K = (S, S_0,\ R,\ L)$ et $K_{\aleph} = (S, S_0\ R_{\aleph},\ L_{\aleph})$ respectivement une structure de Kripke classique et une structure de Kripke floue avec deux propriétés f et $f_{\aleph}$ associées à leur système respectif. La vérification classique consiste à construire l'ensemble

$$\mathbf{e}_K(f) = \{s \in S | \mathbf{e}_{K_s}(f) = 1\},$$

alors qu'en vérification floue on recherche plutôt l'ensemble suivant

$$\mathbf{e}_{K_{\aleph}}(f_{\aleph}) = \{s \in S | \mathbf{e}_{K_{\aleph},s}(f_{\aleph}) > 0\}.$$

On peut remarquer que ces deux notions coïncident en logique binaire ordinaire.

Par ailleurs, il est important de noter que l'ensemble des états est un ensemble *ordinaire* (non-flou) dans la structure de Kripke floue. Il y a certainement moyen de «fuzzifier» l'ensemble des états du système. Mais ce travail n'entre pas dans le cadre de notre mémoire. Sachant cela, il est facile de montrer que la vérification de modèle floue est décidable.

Théorème 6 *La logique floue $\aleph CTL$ telle que décrite dans ce mémoire est vérifiable automatiquement (décidable), c'est-à-dire que pour n'importe quelle structure de Kripke floue, il est possible de construire algorithmiquement en un temps fini l'ensemble $\mathbf{e}_{K_\aleph}(f_\aleph)$, quelque soit la propriété $f_\aleph$ floue donnée.*

Démonstration. Soit $K_\aleph = (S, R_\aleph, L_\aleph)$ une structure de Kripke floue sur un ensemble PA de propositions atomiques. Les formules de $\aleph CTL$ sont toutes construites à partir de PA en composant un certain nombres *fini* d'opérations sur ces formules atomiques. On peut donc parler de la longueur d'une formule sans craindre une infinité de décompositions possibles et permettant de fait une démonstration par récurrence sur la longueur des formules.

Les opérations permises peuvent être limitées à la conjonction floue (min) à la négation floue $(1-\bullet)$ et aux opérations proprement $\aleph CTL$ que sont les dix opérateurs $\mathbf{O}_1\mathbf{O}_2$ avec $\mathbf{O}_1 \in \{\mathbf{A}, \mathbf{E}\}$ et $\mathbf{O}_2 \in \{\mathbf{X}, \mathbf{F}, \mathbf{G}, \mathbf{R}, \mathbf{U}\}$.

Les calculs de $f = \mathbf{Min}(f_1, f_2)$, $f = \mathbf{Max}(f_1, f_2)$ et $f = 1 - f_1$ se terminent en un temps fini par récurrence. De même, en appliquant les transformations de prédicats appropriées, on calcule en un nombre fini d'étapes les ensembles $\mathbf{O}_1\mathbf{O}_2 f$, quelque soit f donnée dans $\aleph CTL$. Ceci achève la démonstration.

CHAPITRE V

CONCLUSION

Dans ce mémoire, nous avons montré qu'il est possible de faire de la vérification de modèles en se basant sur la logique floue, logique comportant une infinité de valences. Dans (2), le travail avait déjà été fait dans le cadre des logiques multivalentes finies.

La démarche suivie ici n'a cependant pas été la même. Dans (2) on définit directement la généralisation multivaluée de CTL. Nous avons plutôt commencé par généraliser CTL^* au cas flou, obtenant $\aleph CTL^*$ avant de définir $\aleph CTL$ comme une sous-logique de $\aleph CTL^*$.

À l'instar de (2) dans le cas multivalué et de (3) dans le cas ordinaire, nous avons ensuite montré que les opérateurs de $\aleph CTL$ étaient tous des points fixes de transformations monotones, démontrant ainsi qu'il est possible de construire algorithmiquement la solution au problème de trouver tous les états d'un système satisfaisant une propriété donnée exprimée dans le langage de $\aleph CTL$.

Cependant, nous n'avons touché ni à l'implémentation, ni aux applications possibles de la théorie. Ce travail reste à faire. Plusieurs autres avenues de recherches pourraient être envisagées à partir du présent contexte. Rapidement, de manière non-exhaustive et en autant que cela n'ait pas été fait auparavant on pourrait, par exemple:

i. «Refaire» ce mémoire avec des opérateurs flous basés sur des disjonctions et des conjonctions moins usuelles que les fonctions *maximum* et *minimum*.

ii. «*Fuzzifier*» les autres logiques $PLTL$, CTL, dite de Dicky, etc.

iii. Établir le contexte approprié en théorie des catégories pour voir que tout le travail que nous venons de faire n'est qu'une «chiquenaude» fonctorielle! En fait, comment pourrait-il

en être autrement? Tout fonctionne tellement bien...

iv. Faire le lien avec la théorie des réseaux. En effet, les graphes orientés sont à la base de ce travail. Le cadre que nous avons examiné pourraient peut-être servir à vérifier certaines propriétés des réseaux. Un examen de la relation $\equiv_x$ nous semble prometteur dans ce contexte.

BIBLIOGRAPHIE

(1) Arnold, A., 1992, *Systèmes de transitions finis et sémantique des processus communicants.* Paris: Masson.

(2) Chechik, M., et Devereux, B., et Easterbrook, S., et Grufinkel, A. 2002, *Multi-Valued Symbolic Model-Checking.* ACM Transactions on Software Engineering and Methodology, 12(4): 371-408, 2003.

(3) Clarke, E. M., Grunberg, O. Jr., et Peled, D. A., 1999 *Model Checking.* Cambridge: The MIT press.

(4) Davey, B.A., et Priestley, H.A., 1990, *Introduction to lattices and order.* Cambridge: Cambridge University Press.

(5) Gacôgne, L., 1997, *Éléments de logique floue*, Paris: Hermes.

(6) Goldejevac, J., 1999, *Idées nettes sur la logique floue*, Lausanne: Presses polytechniques et universitaires romandes.

(7) Gottwald, S., 1993, *Fuzzy Sets and Fuzzy Logic*, Wiesbaden: verlag Vieweg.

(8) Hájek, P. 1998, *Metamathematics of Fuzzy Logic*, Dordrech, Pays-Bas: Kluwer Academic Publisher.

(9) McMillan, K. L., 1993, *Symbolic Model Checking.* Dordrech, Pays-Bas: Kluwer Academic Publisher.

(10) McNeill, D., et Freiberger, P. 1993, *Fuzzy Logic*, New-York: Simon & Schuster.

(11) Muller-Olm, M., et Schmidt, D.A., et Steffen, B.1999, *Model Checking: A tutorial introduction*, In Static Analysis, 12(4): 330-354. Springer-Verlag, LNCS-1694.

(12) Ntambue, R. 1998, *Éléments de logique trivalente*, Louvain-la-Neuve: Bruylant-Academia.

(13) Pnueli, A. Applications of temporal logic to the specification and verification of reactive systems: A survey of current trends. In *Current Trends in Concurrency*, pages 510–584. Springer-Verlag, LNCS-224, 1986.

(14) Schnoebelen, P. 1999 *Vérification de logiciels — Techniques et outils du model-checking.* Paris, Vuibert,.

(15) Stirling, C. 2001, *Modal and Temporal Properties of Processes.* New-York: Springer-Verlag.

(16) Yen, J., et Langari, R. 1999, *Fuzzy Logic*, Upper Saddle River: Prentice-Hall.

(17) Zadeh, L.A., 1971, «Toward fuzziness in computer systems. Fuzzy algorithms and languages.». In *Structure et conception des ordinateurs.*, sous la dir. de Guy Bouliane, p. 9-18. Paris: Dunod.